Explicit Semantic Events Detection and Development of Realistic Applications for Broadcasting Baseball Videos

Wei-Ta Chu[1] and Ja-Ling Wu[1,2]

[1]Dept. of Computer Science and Information Engineering
[2]Graduate Institute of Networking and Multimedia
National Taiwan University, Taipei, Taiwan
wtchu@cmlab.csie.ntu.edu.tw, wjl@csie.ntu.edu.tw

ABSTRACT

This paper presents a framework that explicitly detects events in broadcasting baseball videos and facilitates the development of many practical applications. Three phases of contributions are included in this work: reliable shot classification, explicit event detection, and elaborate applications. At the shot classification stage, color and geometric information are utilized to classify video shots into several canonical views. To explicitly detect semantic events, rule-based decision and model-based decision methods are developed. We emphasize that this system efficiently and exactly identifies what happened in baseball games rather than roughly finding some interesting parts. On the basis of explicit event detection, many accurate and practical applications such as automatic box score generation and game summarization could be built. The reported results show the effectiveness of the proposed framework and demonstrate some research opportunities about bridging the semantic gap for sports videos.

Keywords

Explicit event detection, semantics, sports video, game summarization, highlight extraction.

1. INTRODUCTION

Sports video analysis has attracted much attention due to its potential commercial benefits. Sports games that follow different rules and broadcasting characteristics draw different issues in video analysis. Recently, researchers have developed technologies and applications from different aspects [26]. The

most popular sports such as soccer [6,11,20-22,24,27], American football [2], basketball [14], and baseball [1,7,8,16,17,23,28,29], are widely studied. To the end of providing efficient media access and entertainment functionalities, scene classification [8,29], event detection [7,14,17,20,23-25,28,29], highlight extraction [1-3,16], replay generation [21], or game summarization [6,12,18] have been developed.

Although many studies were proposed to analyze sports video, most of previous works thoughtlessly ignore the real needs of sports audiences, who are the receivers and should be the judgers of analytical results. Generally, a sports fan wishes to know "what really happened in this game?" or "how about my favorite player's performance?" For those who don't have time to see the whole game, a game summary or highlight that consists of the most informative events or exciting parts are attractive. As the famous remark "records are the life of a player" says, practical sports video analysis techniques should deal with "what kind of event occurs," "when and how an event occurs," and "who did it." Explicitly knowing game details is the key factor to make summaries and highlights valuable and reasonable.

Starting from the demands of sports fans, we survey sports analysis techniques in terms of "explicitness" and "comprehensiveness." Explicitness means whether sports events can be exactly detected, such as a "double" in a baseball game or a "three-pointer shot" in a basketball game. Comprehensiveness means whether (almost) all types of events can be detected. For example, thirteen baseball events are defined in Japanese and Taiwanese baseball leagues, and eight common events (goal, shot, foul, corner kick, offside, yellow card, red card, and save) are defined in soccer games. To clarify the novelty and contributions of our work, we remark these issues as follows:

1) Although baseball event detection has been pursued for years, most of them are not comprehensive enough. As compared with other popular sports, such as soccer, tennis, and basketball, more types of events take place in baseball games. That makes achieving comprehensiveness even harder. Zhang

and Chang [28] proposed an event detection method based on caption information, but they only focused on detection of the last pitch and scoring. Han et al. [7] developed a baseball digest system based on maximum entropy method and detected seven baseball events. Nonetheless, the detection performance is not very promising in the reported results. In our work, we deliberate upon detecting all events from the viewpoint of the offensive side and therefore have better understanding of the game content.

2) Recently, explicitness is not easy to be accomplished in other sports video analysis. For example, whether a "fade-away shot" or "dunk" causes a score is hard to be decided, but a basketball fan often likes to see Michael Jordan's fade-away shot or Vince Carter's dunk. Some works such as [17] propose probabilistic approaches to tackle with baseball event detection, but most of them don't provide explicit results. In this work, we develop a rule-based approach assisted with visual analysis to explicitly detect baseball events.

3) The ultimate goal of sports video analysis is to provide users practical applications or well-organized information. Therefore, we turn academic works into realistic applications and evaluate performance by comparing with man-made results or conducting subjective tests.

In this paper, we accentuate our works by carefully tackling with explicitness and comprehensiveness. A systematic framework that comprises reliable shot classification, explicit event detection, and extended applications is proposed. We summarize these processes as follows:

● Reliable shot classification: Color and geometric information are exploited to classify shots into several canonical views. To reliably perform shot classification in different situations (different stadiums, time, or broadcasting channels), several methods to dynamically detect field color and pitcher position are proposed.

● Rule-based event detection and model-based event detection: Official baseball rules are transformed into an efficient rule-based detection module. For the events that cannot be discriminated by simply using baseball rules, model-based detection module is further developed based on elaborately designed game-specific features.

● Extended applications: On the basis of explicit event detection, attractive applications like the ones provides by MLB.com [13] can be automatically developed, such as automatic box score generation or event-on-demand services. Accompanying with audio cues and inherent importance of events, more enjoyable game highlight or summarization could be made.

The rest of this paper is organized as follows. In Section 2, we describe the characteristics of broadcasting baseball videos and propose a framework for explicit event detection. Section 3 provides the details of shot classification. Rule-based and model-based event detection modules are described in Section 4, where the effectiveness of the proposed modules is also addressed. Section 5 states several applications on game abstractions, and discussion and conclusion are given in Section 6.

2. SYSTEM FRAMEWORK

2.1 Characteristics of Baseball Games

An important observation in baseball videos is that all events occur between two consecutive pitch shots. Thus the status changes within this duration give us important clues to indicate what happened in games. The progress of a typical event is: 1) the pitcher releases the ball; 2) the batter hits out the ball; 3) the ball is caught by a fielder (field out) or falls on the ground (hit); 4) a fielder returns the ball to the infield; 5) the camera switches to the pitch view and the pitcher prepares next pitch. Figure 1 illustrates two examples of the game progress. There may be no (duration (a)) or one (duration (b)) event between two consecutive pitch shots.

2.2 Overview of System Framework

Figure 2 shows the system framework, which includes shot classification, event detection, and application development. The target of shot classification stage is to classify video shots into pitch, infield, outfield, or other views. To accommodate to different broadcasting situations, an adaptive field color detection module and pitcher detection module are developed to dynamically extract color and geometric information and to facilitate shot classification.

At the event detection stage, we specially extract caption information in pitch shots. According to baseball rules, the rule-based decision module infers what happened based on the information changes on the caption. However, some event pairs such as 'strikeout vs. field out' cannot be discriminated by simply using rules. The model-based decision module that characterizes shot transition information is further developed to explicitly handle these confused events. After this process, various events are explicitly uncovered.

With the aid of explicit event detection, practical and accurate applications can be developed. To generate more elaborate game abstraction, we consider both the 'informativeness' (content coverage) and 'enjoyability' (perceptual quality) [15] in the summarization and highlight selection process. We design summarization and highlight selection algorithms to produce game abstraction that better matches fans' needs and expectation.

3. SHOT CLASSIFICATION

3.1 Procedure of Shot Classification

Figure 3 shows the process of the shot classification module. Color ranges of the field, including grass and soil, are adaptively determined by a field color determination method. With field color definition, we compute the ratio of field area to the keyframe of each video shot. To avoid some noises derived

from gradual shot transition, the tenth frame from the starting of a shot is selected as its keyframe. The steps of classification are:

1) If the field ratio (FR) is less than the threshold $t1$, the corresponding shot significantly differs from the field and is classified as 'other' view. Typical examples include audience shots or commercial shots.

2) If the field ratio is larger than the threshold $t2$, the corresponding shot is like the field. Based on edge information, an infield/outfield classification module is further developed to distinguish between infield and outfield views.

3) If the field ratio is between $t1$ and $t2$, the corresponding shot is first verified by a pitch shot detection module. If the frame still doesn't conform to the definition of a pitch view, it is further verified by the infield/outfield classification module. Finally, each shot is classified as a pitch, infield, outfield, or other view.

To derive the thresholds $t1$ and $t2$, we gather the statistics of field color ratios from three games and construct their distributions for each canonical view. We model these distributions as Gaussians and find the classification boundaries according to the Bayesian theory. They are finally set as 0.1 and 0.48. Note that although different stadiums or different TV channels bring about significant changes in field color, the presentation of these canonical views is very similar. Therefore, we can feel free to set these thresholds for different games.

3.2 Adaptive Field Color Determination

In baseball games, the field region dominantly occupies the bottom part of video frames. To reliably classify shots, we develop a module to examine the bottom part of video frames and dynamically determine dominant colors. As the game proceeds, this module is triggered periodically to analyze a

video clip and determines the latest color range of the field. In our implementation, it acts on every twenty minutes and analyzes three-minute (3-min) video clips.

All procedures of shot classification are performed in the HLS color space. Three color channels are respectively quantized into thirty equal-interval bins. For each color channel, an integrated histogram is constructed based on the color of pixels in a 3-min video clip (about 5400 frames). We check the integrated histogram and compute the percentage of each bin. If the histogram value is larger than ten percent of total value, the corresponding color range is viewed as the field color. Dominant colors often fall into two ranges, because the baseball field consists of grass and soil.

The assumption of this process is that field colors are the dominant colors in video frames. However, in real broadcasting videos, cameras often switch to the audience or players, or commercials are inserted at inning changes. To remove the influence of these irrelevant shots, we define a default field color range at the beginning of each game. For the bottom part of each frame, we check whether more than forty percent of pixels are 'suspected' field pixels (by the default color definitions). Only the frames with enough suspected field pixels are processed in the field color determination module. The newly determined color range then updates the default field color definition.

3.3 Infield/Outfield Classification

For the keyframe that is largely occupied by field (Figure 3(b)), it is further classified as infield or outfield. The outfield view often contains audience or stadium artifacts and displays high-texture content. Therefore, we use color adjacency histogram [10] to represent edge information and distinguish between infield and outfield views. In our work, the difference between infield and outfield views doesn't affect the performance of event detection, but it may help in game highlight extraction.

3.4 Pitch Shot Detection

For the keyframe whose field ratio is between two thresholds (Figure 3(c)), the spatial layout of field pixels is checked through its horizontal and vertical profiles, as shown in Figure 4. If the keyframe is a pitch view, the field pixels should concentrate only on the bottom part of horizontal profile. Moreover, because the pitcher is always in the left part of a pitch shot, we can find a valley in the vertical profile. To tolerate slight differences between pitch shots in different TV channels, we define a sliding window of 50 pixels with 25-pixel overlapping to go through the left part of the vertical profile (the resolution of testing videos in this work is 352 × 240). If there exists a range whose profile value is less than a threshold (a valley exists), the keyframe is declared to be a pitch view.

Although the field ratio of the case in Figure 3(c) is not as high as that in Figure 3(b), it is also possible to be a field view. The camera may track the ball on the air and doesn't capture most of the field region. Therefore, if no pitcher is detected in the keyframe, it is further confirmed by the infield/outfield classification module.

4. EVENT DETECTION

In this work, we emphasize that we can elaborately perform fine-granularity event analysis based on audiovisual information and baseball rules. For example, if no base is occupied in the ith shot, and the score increases by one but still no base is occupied in the $(i+1)$-th shot, we can infer that a home run (actually a 'solo home run') occurs between these two shots. However, some event pairs such as 'single vs. walk' and 'strikeout vs. field out' cannot be discriminated by simply using the rule-based decision method. We further take the contextual shot information into account and propose a model-based decision method to strive for explicit event detection. The most important contribution of this work is that we propose a systematic method to seamlessly integrate domain knowledge (baseball rules) with

computational characteristics (audiovisual features). By this way, we thoroughly explore baseball games and achieve "comprehensiveness" and "explicitness" in event detection.

4.1 Rule-based Event Detection

In broadcasting videos, informative caption data include 'number of out, 'number of score', and 'base-occupation situation'. Each effective baseball event leads to changes of this information, such as 'home run' increases the score, 'strikeout' increases out, and 'hit' and 'walk' change the base-occupation situation. Therefore, we can check the information changes on the caption and accomplish efficient implementation for event detection.

4.1.1 Caption Feature Extraction

Caption information is often displayed as two types: text (such as number of score) or symbol (such as number of out and base occupation). For text information extraction, three steps are included:

● Character pixel determination: Characters often have higher intensity as compare to the background. The pixel that has high intensity is viewed as a character pixel.

● Construct character template vectors: Given a region, a 13-dimensional Zernike moment [9] is extracted to represent character pixels' characteristics. For each number, e.g. two, we select a 30-second video clip (about 900 frames) as the training data. The Zernike moments extracted from video frames are then averaged to construct the character template.

● Character recognition: Given a test vector, it is compared with all trained character templates in terms of vector angle. The test vector is recognized as i if it has the smallest included angle to the ith character template.

For symbol information, we just employ the intensity-based approach similar to character pixels segmentation. In the pre-indicated region, the base-occupation situation is displayed according to whether the corresponding base is highlighted or not.

In the duration between two consecutive pitch shots, the changes of number of out, number of score, and base-occupation situation are jointly considered in event detection. They are:

- $o_{i,i+1}$, the difference of outs between the ith and the $(i+1)$-th pitch shots, where $o_{i,i+1} \in \{0, 1, 2\}$. We don't deal with the situation of $o_{i,i+1} = 3$ because, in almost all TV channels, commercials are instantaneously inserted when there are three outs.

- $s_{i,i+1}$, the difference of scores between the ith and the $(i+1)$-th pitch shots, where $s_{i,i+1} \in \{0, 1, 2, 3, 4\}$. The case of $s_{i,i+1} = 4$ denotes the occurrence of a home run with four scores (the so-called 'grand slam').

- b_i and b_{i+1}, the base-occupation situations in the ith and the $(i+1)$-th pitch shots, where b_i and b_{i+1} $\in \{0, 1, ..., 7\}$. The number of occupied bases at these two shots (n_i and n_{i+1}) are calculated. And to catch the difference of base occupation situations, the value of $b_{i,i+1}$ $(= b_{i+1} - b_i)$ is also considered. The meanings of feature values of b_i and n_i are listed in Table 1.

4.1.2 Feature Filtering

The features described above are concatenated as a vector $f_{i,i+1}$ to represent the game progress. However, many cases are illegal in baseball games. We should filter out the illegal features and identify the events implied by legal features.

When an event occurs, there may be one or no batter reaching a base, and the runners (the players who occupy the bases) would be still at bases or out or reach the home plate to get scores. Therefore, when a legal event occurs, one of the following situations might take place:

1-1) The batter is out, and a runner reaches the home plate to get score. In this case, $o_{i,i+1} + s_{i,i+1} + n_{i,i+1}$
$= 1 + 1 + (-1) = 1.$

1-2) The batter is out, and the runner(s) reach some bases (but not the home plate). In this case, $o_{i,i+1} +$
$s_{i,i+1} + n_{i,i+1} = 1 + 0 + 0 = 1.$

1-3) The batter is out, and the runner(s) also get outs. In this case, only double play events are considered, and $o_{i,i+1} + s_{i,i+1} + n_{i,i+1} = 2 + 0 + (-1) = 1$. Triple play is a special case and is especially stated in Section 4.1.3.

2-1) The batter reaches a base, but one runner gets an out. In this case, $o_{i,i+1} + s_{i,i+1} + n_{i,i+1} = 1 + 0 + 0 = 1$.

2-2) The batter reaches a base, but two runners get outs. In this case, $o_{i,i+1} + s_{i,i+1} + n_{i,i+1} = 2 + 0 + (-1) = 1$.

3-1) The batter reaches some base (but not the home plate), and some runners reach the home plate to get scores. No matter how many runners getting scores, $n_{i,i+1} + s_{i,i+1} = 1$. For example, assume that the second and the third bases are occupied in the ith pitch shot. The batter hits a double and reaches the second base, and both two runners reach the home plate to get two scores. The information change is $(n_{i+1} - n_i) + s_{i,i+1} = (1\text{-}2) + 2 = 1$. Therefore, in all possible situations of this case, $o_{i,i+1} + (s_{i,i+1} + n_{i,i+1}) = 0 + 1 = 1$.

3-2) The batter reaches the home plate (i.e. a homerun), and of course all runners reach the home plate to get scores. In all possible situations of this case, $o_{i,i+1} + (s_{i,i+1} + n_{i,i+1}) = 0 + 1 = 1$.

4-1) The batter is neither out nor reaching a base, and one or more runners advance to next bases (but not getting score). In this case, $o_{i,i+1} + s_{i,i+1} + n_{i,i+1} = 0 + 0 + 0 = 0$.

4-2) The batter is neither out nor reaching a base, and one runner steals the home plate. In this case, $o_{i,i+1} + s_{i,i+1} + n_{i,i+1} = 0 + 1 + (-1) = 0$.

4-3) The batter is neither out nor reaching a base, and runner(s) are neither out nor scoring nor reaching

next bases. In this case, $o_{i,i+1} + s_{i,i+1} + n_{i,i+1} = 0 + 0 + 0 = 0$.

4-4) The batter is neither out nor reaching a base, but one or more runners are out (due to "caught stealing" or "pick-off", for example). In this case, $s_{i,i+1} + (o_{i,i+1} + n_{i,i+1}) = 0 + 0 = 0$.

According to these observations, a general decision rule for legal features can be mathematically expressed as:

$$f_{i,i+1} = \begin{cases} legal, & \text{if } (n_{i,i+1} + s_{i,i+1} + o_{i,i+1}) = 0 \text{ or } 1, \\ illegal, & \text{otherwise.} \end{cases} \tag{1}$$

The value of $(n_{i+1} + s_{i,i+1} + o_{i,i+1})$, denoted as $\alpha_{i,i+1}$, indicates whether the batter changes ($\alpha_{i,i+1} = 1$) or not ($\alpha_{i,i+1} = 0$).

Furthermore, according to the baseball rules, no runner can go back to the previous base. We check the base-occupation situations in two consecutive pitch shots and filter out this kind of illegal features. For example, it would not happen if $b_i = 2$ and $b_{i+1} = 1$ in case of $s_{i,i+1} = 0$ and $o_{i,i+1} = 0$. (It's impossible that the occupied base is back in case of no score and no out.)

4.1.3 Event Identification

Given a legal feature vector, we can view the process of event identification as classifying it into a subset, which represents one baseball event. The given feature vector is first classified as one of the four types of events by checking whether the batter changes ($\alpha_{i,i+1} = 0$ or 1) and whether the number of out ($o_{i,i+1}$) increases. The baseball event taxonomy is illustrated in Figure 5. Thirteen events are considered in this work: single (1B), double (2B), triple (3B), home run (HR), stolen base (SB), caught stealing (CS), field out (AO), strikeout (SO), base on ball (Walk, BB), sacrifice (SAC), sacrifice fly (SF), double play (DP), and triple play (TP). Although they still don't cover all events in baseball games, they explicitly state what happens in a game and greatly expand the visibility of baseball videos.

The event identification process can be conceptually modeled as a decision tree, as shown in Figure 6, and events are determined at leaves after tracing the tree. We construct the decision tree based on baseball knowledge and elaborately include most events in leaves for achieving comprehensive event detection. The general idea of tracing this tree can be described as follows:

(1)　First, we check $o_{i,i+1}$ to detect whether the unknown event causes an out or not.

(2)　According to $n_{i,i+1}+s_{i,i+1}$, check whether a new runner occupies a base or someone scores.

(3)　Then we check base-occupation situation (b_{i+1} or $b_{i,i+1}$) to determine what really happened.

Note that the event of triple play (TP) is a special case and is not included in Figure 6. It's a very unusual event and is detected by a heuristic rule that is beyond the constraint of the equation 1:

If more than two bases are occupied and nobody out in the ith pitch shot, and nobody out, no score changes, and no base is occupied in the ($i+1$)-th pitch shot ($o_{i,i+1}=0$, $n_{i,i+1}=-2$ or -3, $s_{i,i+1}=0$), a triple play would occur.

The rule-based process effectively detects most events in baseball games. However, some event pairs such as 'single vs. walk' and 'strikeout vs. field out' lead to the same information changes on caption and cannot be explicitly discriminated by simply using rules. These kinds of 'confused' events can be categorized as four types, as shown in Table 2. To make the event detection process more explicitly, we develop a model-based approach that adopts contextual shot information and elaborate the detection results. We primarily deal with the cases of 'single vs. walk' and 'strikeout vs. field out' because other confused situations rarely happen.

4.2 Model-based Event Detection

Shot transition information and its temporal duration often provide clues for event identification. For example, as a single occurs, the camera switches to the field to show the action of fielder. On the other hand, as a base on ball occurs, close-up on the pitcher or the batter is often captured. In the model-based

event detection, we jointly consider information of shot transition, temporal duration, and motion magnitude for discriminating events that are implicitly hidden after rule-based event detection.

4.2.1 Shot Context Features

According to the observation of broadcasting styles, we propose the following features to describe event characteristics. Note that these features are extracted within the duration from the end of previous effective event to current pitch shot, as shown in Figure 7.

- *ConsecutivePF*: indicating whether a field view displayed immediately after the last pitch view. If the batter hits out the ball, this kind of shot pair occurs and indicates higher probability of the occurrence of 'single' or 'field out'. In Figure 7, the last pitch view is at the third shot, and the shot pair occurs at the third-fourth shots to indicate *ConsecutivePF* = 1.

  The first field shot right after the last pitch shot plays an important role in extracting shot context features and is particularly defined as the *pivot shot*. If there is no field shot within this duration, the last shot of this duration is defined as the pivot shot.

- *PitchBeforeFieldView*: indicating how many pitch views before the pivot shot. In general, more pitch shots occur before the pivot shot in the events of 'walk' and 'strikeout', because the pitcher has to pitch at least four or three balls before they take place. In the above example, the batter hits the ball at the second pitch (Figure 7(3)), and therefore, *PitchBeforeFieldView* = 2.

- *DiffPitchField*: indicating the time difference between the last pitch shot and the pivot shot. If the batter doesn't hit out the ball, i.e. *ConsecutivePF* = 0, *DiffPitchField* is often larger in 'walk' and 'strikeout' cases than that in 'single' and 'field out' ones.

- *FieldDuration*: indicating the time duration of the pivot shot. When the ball is hit out, the duration of field shot is often short because the fielder should deal with the ball as soon as possible to prevent extra base hit. In Figure 7, *FieldDuration* = 1237-1151 = 86 frames.

● *Motion*: indicating the motion magnitude of the pivot shot. When the ball is hit out, the camera tracks the ball or the fielder and demonstrates higher motion. Therefore, higher motion is often derived from 'single' or 'field out' events, and lower motion is derived from 'walk' or 'strikeout' cases.

4.2.2 Modeling

All the shot context features are normalized to the range [0,1] before training or testing. We manually selected twenty training sequences, ten of them are 'single' and another ten sequences are 'walk', from the same TV channel to construct a 'single-walk' classifier. K-nearest neighbor modeling is implemented for each classifier due to its simplicity. Through the rule-based decision described in Section 4.1, the sequences decided as '1B or BB' are further discriminated by the classifier. The shot context features obtained from the suspected sequence are then classified as a 'single (1B)' or 'walk (BB)' event by the k-nearest neighbor algorithm. The same process is applied to discriminate 'field out' and 'strikeout'. In this work, k is set as 8 for classification accuracy and efficiency.

4.3 Results of Event Detection

Most of the evaluation data in this paper are taken from the games of Chinese Professional Baseball Leagues (CPBL) [4], which are broadcasted by Videoland Sport Channel [19]. Four different broadcasting games with total length about ten hours are used. Three of them are CPBL games in year 2004 (CPBL1 and CPBL2 in Table 3) and year 2005 (CPBL3), and one of them are from MLB (Tigers vs. Yankees at 2005/5/26). They are recorded directly from TV, and the commercials are not intentionally filtered out. Because the proposed framework only employs the caption information in pitch shots and shot transition information, commercials that are often classified as "other" shots would not degrade the detection performance. This flexibility makes the proposed approach more practical in developing a system that real-time analyzes broadcasting videos and immediately provides analytical results right after games.

For shot classification, we briefly evaluate the performance by checking four half innings in four different games (298 shots), where averagely 91.6% accuracy can be achieved. If we only consider pitch shot detection, which is the most indicative shot type in this work, there are actually 53 pitch shots among the evaluation data. The proposed shot classification process correctly detects 52 shots in 59 detected cases. Thus the precision and recall rates are 98% and 88%, respectively. Table 3 shows the detection results of six frequently occurred events in terms of precision and recall rates. The term 'HB' denotes the events of 'single' or 'walk', and 'O' denotes 'strikeout' or 'field out'. The numbers in parentheses (n_1/n_2) in each row denote the counts of events to calculate precision and recall. Overall, we obtain very promising results in detecting most events. At least 0.85 of precision rate and 0.9 of recall rate can be achieved. The detection performance in MLB is slightly worse because of worse shot classification and character recognition accuracy deriving from poorer video quality. Note that although only common events are shown in Table 3, other rare events could also be correctly detected by the proposed method. For example, the only 'triple' event in CPBL2 and the only 'caught stealing' event in CPBL3 are both correctly detected.

Table 4 shows the results of discriminating confused events, i.e. 'HB' and 'O' events in Table 3. The discrimination performances of single, walk, and field out are satisfactory, while that in strikeout is still needed to be improved. One reason for the worse strikeout/field out discrimination result may be the imbalance testing dataset. The number of strikeout is much fewer than that of field out (there are totally 39 strikeouts and 114 field outs in these four games), and the errors have more effects on the strikeout classification results. To combat this problem, we can jointly consider likelihood and priori probability of strikeout (field out) so that the influence of priori probability is included. Considering the statistical characteristics would be our future direction to develop more reliable classifiers.

Overall, the proposed framework achieves satisfactory performance without being affected by game variations. An on-line system demo is in the "explicit event detection" part at http://www.cmlab.csie.ntu.edu.tw/~wtchu/baseball/index.html. We present sample results of event detection and provide a prototypal event-on-demand service on the web.

Note that some byproducts can be obtained after event detection. "Runs battered in (RBI)," which denotes number of scores as a direct result of an event, can be calculated from the changes of scores. "Left on base (LOB)," which denotes the total number of runners who did not score when the batter made an out, can be calculated by checking how many bases were occupied before the batter was out. This information represents the effectiveness of events and can be good indicators for game abstraction.

5. EXTENDED APPLICATIONS

After event detection, event types in different durations have been determined. To develop practical applications, we extract an event from the last pitch to the first pitch of the next event (e.g., from the third shot to the nth shot in Figure 7). This method reserves the most significant parts and provides efficient event presentation. However, the last events in each half inning are often very long because commercials would be inserted. Therefore, we limit the length of each event (not only the last event in a half inning) no more than forty seconds to prevent very long events. This constraint is widely applied in the following applications. For example, in the event-on-demand service, if the last pitch starts at time T (seconds), users can access this event from T to T+40 seconds.

5.1 Automatic Generation of Box Score

Baseball games have fixed and strict structures. One baseball game usually has nine innings, each inning has two half innings, and each half inning has three outs. Moreover, the batter enters the field and invokes events sequentially. Therefore, by integrating the results of explicit event detection with the given lineup, we can gather the performance of each player and generate box score. The box score

facilitates users to realize a game at a glance. Users can click events of interest and view the corresponding video clips immediately. The event-on-demand service is widely adopted on the internet and is one of the most popular ways to access a game. With explicit event detection, we are able to achieve automatic box score generation and provide a very useful service as done by [13].

5.2 Automatic Game Summarization

A reasonable game summary should include the clips with scoring and the progress of effective offenses, like the sequence of "a single, a sacrifice bunt, and a double" that causes a score. We argue that approaches based on low-level features and probabilistic methods cannot accurately achieve the requirements without explicit event detection. With the aid of explicitness, we can develop superior summarization modules to appropriately represent the content of games.

5.2.1 Significance Degree of Events

To maintain informativeness of a game summary, we give different significance degrees to different types of events according to their contributions. Babaguchi et al. [2] propose an idea to define the significance degrees of events for American football, while they perform game summarization from existing text-based game logs rather than the results of automatic event detection. We follow similar ideas and modify the definition of significance especially for baseball videos. Five levels of significance degrees are defined in the following:

● Rank 1: state change events. Only three states exist in team sports: "the two teams tie", "team A leads", and "team B leads". The events that cause one team to score and change the current state into a different state are called as state change events. They are the indicators of winning pitcher and winning RBI, and are given the greatest significance.

● Rank 2: hits with RBIs, steals with score, or sacrifice fly. Hits with RBIs, no matter they are single, double, or home run, change score and directly affect the result of the game. They also indicate the

effectiveness of hits. Moreover, the very rare case that a runner steals the home plate and scores is also defined as a rank-2 event.

● Rank 3: hits without RBI, walks, steals without score, double play, or triple play. Although no score is obtained, the number of hits is concerned with a player's batting average. Moreover, steals without score have similar effects and are defined as rank-3 events. As regards to double play and triple play, we categorize them as rank-3 events to emphasize the impacts of outstanding defenses.

● Rank 4: strikeouts with LOBs, field outs with LOBs, or sacrifice. These kinds of outs show that the batter fails to help teammates score. Larger LOB indicates more negative influence when a play makes an out.

● Rank 5: strikeouts without LOB, field outs without LOB, or caught stealing. Normal outs generally cover more than half of cases and give the least significance.

Note that the ranking definitions described above are determined according to baseball conventions. They are commonly accepted in evaluating the importance of baseball events. Actually, different definitions can be made to facilitate generating different kinds of summaries. For example, a user can increase the significance degree of the events invoked by his favorite players or teams.

5.2.2 Selection of Summarization

We provide various summaries that have different lengths and information coverage. The most compact summary can be generated by concatenating rank-1 events, while a richer summary can be formed by collecting rank-1, rank-2, and rank-3 events. In addition to event rank that is defined for each isolated event, context of events in a half inning should also be considered in event selection. In baseball games, there may be a chain of events to result in scoring. Although some of these events may be in lower rank, the chain of events should be collected together to maintain the completeness of summary. For example, in Figure 8, a chain of double, strikeout, and single events occur and finally cause scoring. The single

leads to a score because the second base is occupied. Hence it's no doubt that both the double and single events should be collected in the summary. Moreover, the audience usually expects the player to have a good play when some bases are occupied. The result of his play impresses the audience, no matter it's a good play causing RBI or a bad play causing LOB. Therefore, we also take account of the context of events and collect the strikeout event in summary. On the other hand, if only one rank-3 event occurs alone (no other events with ranks $\leqq$ 3), it should be ignored because fragmentary hits don't cause effective results.

On the basis of this ideas, three levels of game summaries are automatically generated:

- Level 1: Only the rank-1 and rank-2 events are collected. This level of summary contains the most compact results.

- Level 2: Basically, only events with ranks 1~3 are collected. Rank-3 events and rank-4 events are considered to be discarded or added through checking event context:

  - ◆ Rank-1 and rank-2 events are definitely picked as the summary.
  - ◆ Check each rank-3 event E_i.

    If both the ranks of the (i+1)-th and (i+2)-th (r_{i+1} and r_{i+2}) events are less than 4, pick them all as the summary.

    If r_{i+1} < 3 and r_{i+2} = 5, just pick the ith and (i+1)-th events as the summary.

    If r_{i+1} = 4 and r_{i+2} = 5, ignore all the ith, (i+1)-th, and (i+2)-th events.

  Note that rank-3 and rank-5 events would not occur successively because of the inherent baseball rules.

- Level 3: All events with ranks 1~5 are collected. This level of summary contains the most complete content of a game. With the length constraint described in the beginning of Section 5, this level of summary still contains informative videos clips, in which most commercials or irrelevant clips are eliminated.

Note that the temporal relationships between events should be maintained, because a summary formed by disordered events is meaningless. Therefore, different ranks of events may be interlaced. Moreover, the final length of summary depends on the content of a game. If two teams have a keen competition, the length of summary will be larger due to more rank-1 and rank-2 events. Table 5 shows the lengths of summaries from two different competitions.

5.2.3 Evaluation of Summarization

To evaluate the effectiveness of the proposed summarization method, we compare automatic game summarization with man-made condensed games, which are taken from a sport TV station [19]. Events in the man-made condensed games are selected by professional sports reporters. Although the selected events may not be exactly the same from different reporters or different TV channels, they can be viewed as good references for evaluation. Two indicators are calculated:

$$\text{Precision}=\frac{N_c}{N_s}, \qquad \text{Recall}=\frac{N_c}{N_m},$$

where N_S is the number of events in the automatic summary, N_m is the number of events in the man-made summary, and N_c is the number of events in both summaries.

In our experiments, level-2 summaries meet audience's expectation the most and are close to the condensed games made by professional sports reporters. Therefore, we compare them with the man-made summary and show the performance in Table 6. The values n_2/n_1 in each inning denote that n_1 events are collected by the proposed process, and among them, n_2 events are in the man-made summary. To demonstrate the effectiveness of the summarization selection mechanism, we juxtapose the results with and without considering context of events. In the game of "Lions vs. Whales," the precisions of with-context and without-context summaries are 0.886 and 0.875, and the corresponding recalls are 0.939 and 0.848. Although the precision rates are similar in two approaches, the with-context summary

clearly has superior performance in recall rates. This result is expectable, because the summary without considering event context often misses the rank-4 event that is between two rank-3 events. To maintain the completeness of a game summary, professional sports reporters prefer to include this event in summary. The performance difference aggravates in the game of "Bulls vs. Lions". The summary with context consideration has significantly superior performance in both precision and recall rates. Details of summary comparison can be seen in the "game abstraction" part at http://www.cmlab.csie.ntu.edu.tw/~wtchu/baseball/index.html.

5.3 Automatic Highlight Generation

Another attractive application is game highlight extraction. To maintain entertaining functionalities within short time duration, highlight extraction poses different concerns from summarization. It is evident that effective events such as state change events or hits with RBIs should be highlighted. In addition, beautiful defense play such as diving catch or caught stealing should also be highlighted, although they just cause a normal field out. In highlight extraction, we integrate the impacts of event ranks, audio energy dynamics, and occurrence time to generate a game highlight that well retains 'enjoyability' of a game.

5.3.1 Significance Degree of Events

5.3.1.1 Rank-based Significance

For the requirement of highlight, we slightly modify the definition of event rank. Double play, triple play, and caught stealing events are categorized as rank-2 events to cover important defense. The rank-based significance degree S_r $(0 \leq S_r \leq 1)$ of the ith event E_i is quantitatively defined as

$$S_r(E_i) = 1 - \frac{r_i - 1}{5} \cdot \alpha, \tag{2}$$

where r_i $(1 \leq r_i \leq 5)$ denotes the rank of the ith event, and α $(0 \leq \alpha \leq 1)$ is the parameter controlling the weight of event rank.

5.3.1.2 Time-based Significance

The events occurring at the latter stage of games are usually more attractive to users, especially when two teams tie or have slight score difference. The time-based significance S_t $(0 \leq S_t \leq 1)$ is defined as

$$S_t(E_i) = 1 - \frac{N - I(E_i)}{N} \cdot \beta, \tag{3}$$

where $I(E_i)$ denotes the inning in which the event E_i occurs, N is the number of total innings in a game (usually nine innings), and β $(0 \leq \beta \leq 1)$ is the parameter controlling the weight of occurrence time.

5.3.1.3 Audio-based Significance

The anchorperson often comments excitedly and the audience cheers loudly when a beautiful play or an important hit occurs. We analyze audio energy dynamics to show how they react to an event. Audio energy dynamics can be broadly classified into regions of attack, sustain, decay, and silence [5]. We focus on detecting attack because it indicates the occurrence of an exciting event.

We evaluate the envelope of power spectrum and only concentrate on how audio energy increases:

$$\begin{aligned}
&if \ e_k - \text{mean}(e_{k-w}, ..., e_{k-1}) > 0 \\
&\quad d_k = e_k - \text{mean}(e_{k-w}, ..., e_{k-1}), \quad k = 1, 2, ..., M, \\
&else \ d_k = 0,
\end{aligned}$$

where e_k denotes the average energy of the kth audio segments, and M denotes the number of audio segments within the duration of an event. Energy difference d_k is calculated by subtracting average energy of previous w segments from e_k. Each audio segment is of length one second, and w is set as four in this work. The maximum energy difference within the event duration is chosen and is quantized into one to five:

$$D_i = \text{Quantize}(\max(d_k)). \tag{4}$$

Accordingly, the audio-based significance S_a $(0 \leq S_a \leq 1)$ is defined as

$$S_a(E_i) = 1 - \frac{5 - D_i}{5} \cdot \gamma, \tag{5}$$

where γ $(0 \leq \gamma \leq 1)$ is the parameter controlling the weight of audio cues.

By combining the impacts of event rank, occurrence time, and audio energy dynamics, the integrated significance degree $S(E_i)$ $(0 \leq S \leq 1)$ is given by

$$S(E_i) = S_r(E_i) \cdot S_t(E_i) \cdot S_a(E_i). \tag{6}$$

Different highlights could be obtained by changing the weighting parameters of α, β, and γ. In our experiments, we set α, β, and γ as 0.5, 0.2, and 0.3, respectively.

5.3.2 Highlight Selection Algorithm

In general, highlight selection can be formulated as a knapsack problem. That is, given segments of different significance and lengths, find the most significant set of segments that fit in a knapsack of fixed length. In our work, we implement a greedy approach to select highlighted events. By considering the time limitation given by the user and event context, the highlight selection algorithm is as follows:

Input: the user-defined highlight length T and the set of events E in the game.

Output: the set of highlighted events A.

HIGHLIGHT_SELECTION(T, E)
1 $A \leftarrow \varnothing$
2 sort E into nonincreasing order by significance degrees
3 for each $e_i \in E$
4   do if length of $(A \cup \{e_i\}) < T$
5   then $A \leftarrow A \cup \{e_i\}$
6   SMOOTH(A)
7 return A

Similar to the context idea in automatic summarization, adjacent relationships between highlighted events are considered in the SMOOTH process. For three adjacent events A-B-C, if both A and C events

are selected as highlight, B is also selected to maintain the completeness of a game highlight. Finally, the selected events are sorted by the occurrence time to maintain temporal coherence.

5.3.3 Evaluation of Highlight

The goal of a highlight is to "excite" the audience. However, someone would like to see scores; some others would like to see their favorite players; and some others would like to see nice defense plays. It's hard to generate a highlight that satisfies all users, and no ground truth can be the standard for evaluation. Therefore, we invited 24 persons, including 21 males and 3 females, to perform subjective experiments based on highlights extracted from two games. We impose an assumption on the subjects: none of the subjects saw these games before. This assumption is reasonable for simplification purpose, because we cannot expect every subject affords to spend more than six hours to see two baseball games.

The experimented scenario is set to be event-based. Because of the assumption, we didn't ask subjects 'Does the game highlight contain the most highlighted parts of this game?' Instead, we request subjects to evaluate each selected event: "Do you think this event is exciting?" This evaluation somehow represents the 'accuracy' of the proposed highlight selection method. Because an event's significance sometimes depends on the effectiveness of the succeeding events, we present multiple events together if they are in the same half inning. After presentation of one half inning, the subjects give one opinion score (from one to five, indicating from bad to excellent) to each selected event to judge whether it's a highlight part.

The selected highlight events and their corresponding meanings are listed in Tables 7 and 8. Table 9 shows the subjective results of highlights with different lengths. Eleven events and eight events are selected to construct 7-min and 5-min highlights, respectively. From Table 9, highlights from both games satisfy users and get average score larger than 3.3. The shorter highlight getting higher score indicates that the proposed significance degree modeling positively captures the characteristics of

highlights. Moreover, human's subjective satisfaction is slightly affected by the competitive content of games. In "Bulls vs. Lions", three home runs occurred and two teams have a keen competition. On the other hand, the team "Lions" dominates in "Lions vs. Bears", and the game presents flat content. Therefore, the events selected in "Bulls vs. Lions" often excite the subjects more and get higher scores. Recently, game highlights are popular materials for representing game content in sports news or on-line entertainment services [13]. A man-made highlight, e.g. the highlight reel of an MLB game, consists of video shots elaborately edited and remarkable comments. This kind of game highlight impresses the audience while it requires lots of professional equipments and working time. In this work, we present an automatic highlight selection method that provides satisfactory highlights and is free from user intervention. For personalization purpose in digital home environment, users can adjust the weights with respect to different factors to generate different flavors of highlights.

6. DISCUSSION AND CONCLUSION

We have presented a complete strategy to perform explicit event detection and develop practical applications for broadcasting baseball videos. Adaptive field color determination and dynamic pitcher position detection are devised to make shot classification more reliable and general. Then, the rule-based and model-based decision methods are integrated to explicitly detect thirteen baseball events. Official baseball rules are transformed into a decision tree in the rule-based decision module, while the context of shots is considered in the model-based decision module. Finally, on the basis of explicit event detection, automatic game summarization and highlight selection are implemented to preserve "informativeness" and "enjoyability" within short duration. Elaborate design of the significance degree of events and various evaluations are presented. The proposed approaches automate broadcasting baseball video analysis and facilitate various applications.

In summary, the major contribution of this work is that we propose a systematic method to explicitly and comprehensively analyze baseball videos. We believe that the analytical results and developed applications are more realistic to users. It's arguable that this work specifically concentrates on the domain of baseball games and is not intuitive to be extended. However, utilizing domain knowledge more exhaustively leads to more practical functionalities. We engage in thorough studies of baseball video analysis and report results that appropriately match the demands of most users. From all of our survey, the proposed method best exploit official rules in event detection and game abstraction, and provides the most comprehensive and practical results in baseball video analysis.

Although we explicitly detect various events in baseball games, some special events are still not included. For example, player substitution and defense error are subtle or are determined subjectively by the umpire. To further scrutinize game content, we would include a speech recognition module to detect some keywords such as error or substitution in commentator's speech. Furthermore, there is still space to improve the highlight selection method. Various applications with different purposes and for different services can be further developed.

7. REFERENCES

[1] Y. Ariki, M. Kumano, and K. Tsukada, "Highlight scene extraction in real time from baseball live video," in Proc. of ACM Int. Workshop on Multimedia Information Retrieval, pp. 209-214, 2003.

[2] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, "Personalized abstraction of broadcasted American football video by highlight selection," IEEE Trans. on Multimedia, Vol. 6, No. 4, pp. 575-586, 2004.

[3] M. Bertini, A. Del Bimbo, and W. Nunziati, "Highlights modeling and detection in sports videos," Pattern Analysis and Applications, Vol. 7, No. 4, pp. 411-421, 2004.

[4] Chinese Professional Baseball League, http://www.cpbl.com.tw

[5] C. Dorai and S. Venkatesh, Media Computing: Computational Media Aesthetics, Kluwer Academic Publishers, 2002.

[6] A. Ekin, A.M. Tekalp, and R. Mehrota, "Automatic soccer video analysis and summarization," IEEE Trans. on Image Processing, Vol. 12, No. 7, pp. 796-807, 2003.

[7] M. Han, W. Hua, W. Xu, and Y. Gong, "An integrated baseball digest system using maximum entropy method," in Proc. of ACM Multimedia Conference, pp. 347-350, 2002.

[8] W. Hua, W. Han, and Y. Gong, "Baseball scene classification using multimedia features," in Proc. of ICME, pp. 821-824, 2002.

[9] A. Khotanzad and Y.-H. Hong, "Invariant image recognition by Zernike moments," IEEE Trans. on PAMI, Vol. 12, No. 5, pp. 489-497, 1990.

[10] H.Y. Lee, H.K. Lee, and Y.H. Ha, "Spatial color descriptor for image retrieval and video segmentation," IEEE Trans. on Multimedia, Vol. 5, No. 3, pp. 358-367, 2003.

[11] R. Leonardi, P. Migliorati, and M. Prandini, "Semantic indexing of soccer audio-visual sequences: a multimodal approach based on controlled Markov chains," IEEE Trans. on Circuits and Systems for Video Technology, Vol. 14, No. 5, pp. 634-643, 2004.

[12] B. Li, J.H. Errico, H. Pan, and I. Sezan, "Bridging the semantic gap in sports video retrieval and summarization," Journal of Visual Comm. and Image Representation, Vol. 15, pp. 393-424, 2004.

[13] Major League Baseball, http://www.mlb.com

[14] S. Nepal, U. Srinivasan, and G. Reynolds, "Automatic detection of goal segments in basketball videos," in Proc. of ACM Multimedia Conference, pp. 261-269, 2001.

[15] C.-W. Ngo, Y.-F. Ma, and H.-J. Zhang, "Video summarization and scene detection by graph modeling," IEEE Trans. on Circuits and systems for Video Technology, Vol. 15, No. 2, pp. 296-305, 2005.

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

[17] H.-C. Shih and C.-L. Huang, "A semantic network modeling for understanding baseball video," in Proc. of ICASSP, Vol. 5, pp. 820-823, 2003.

[18] D. Tjondronegoro, Y.-P. P. Chen, and B. Pham, "Integrating highlights for more complete sports video summarization," IEEE Multimedia, Oct.-Dec., pp. 22-37, 2004.

[19] Videoland Sports Channel, http://sport.videoland.com.tw/

[20] L. Wang, M. Lew, and G. Xu, "Offense based temporal segmentation for event detection in soccer video," in Proc. of ACM Int. Workshop on Multimedia Information Retrieval, pp. 259-266, 2004.

[21] J. Wang, C. Xu, E. Chng, K. Wan, and Q. Tian, "Automatic replay generation for soccer video broadcasting," in Proc. of ACM Multimedia Conference, pp. 32-39, 2004.

[22] L. Xie, P. Xu, S.-F. Chang, A. Divakaran, and H. Sun, "Structure analysis of soccer video with domain knowledge and hidden Markov models," Pattern Recognition Letters, Vol. 25, No. 7, pp. 767-775, 2004.

[23] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T.S. Huang, "Audio events detection based highlights extraction from baseball, golf and soccer games in a unified framework," in Proc. of ICASSP, Vol. 5, pp. 632-635, 2003.

[24] H. Xu and T.-S. Chua, "The fusion of audio-visual features and external knowledge for event detection in team sports video," in Proc. of ACM Int. Workshop on Multimedia Information Retrieval, pp. 127-134, 2004.

[25] G. Xu, Y.-F. Ma, H.-J. Zhang, and S. Yang, "A HMM based semantic analysis framework for sports game event detection," in Proc. of ICIP, Vol. 1, pp. 25-28, 2003.

[26] X. Yu and D. Farin, "Current and emerging topics in sports video processing," in Proc. of ICME, pp. 526-529, 2005.

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

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

[29] D. Zhong and S.-F. Chang, "Real-time view recognition and event detection for sports video," Journal of Visual Comm. and Image Representation, Vol. 15, pp. 330-347, 2004.
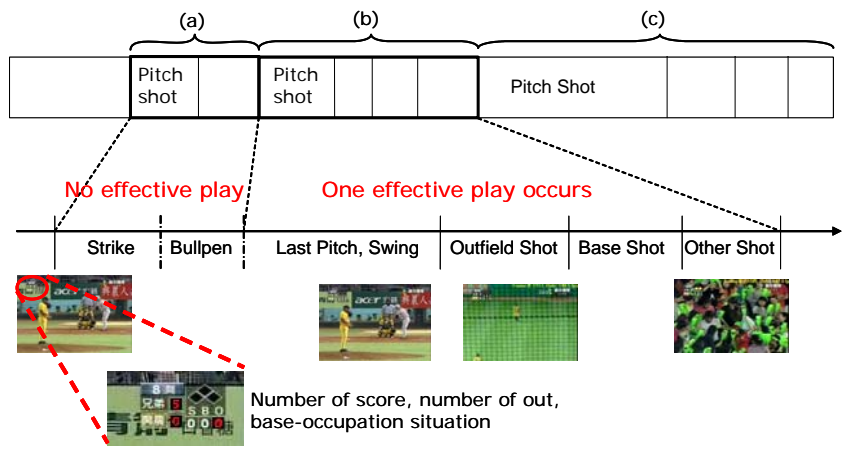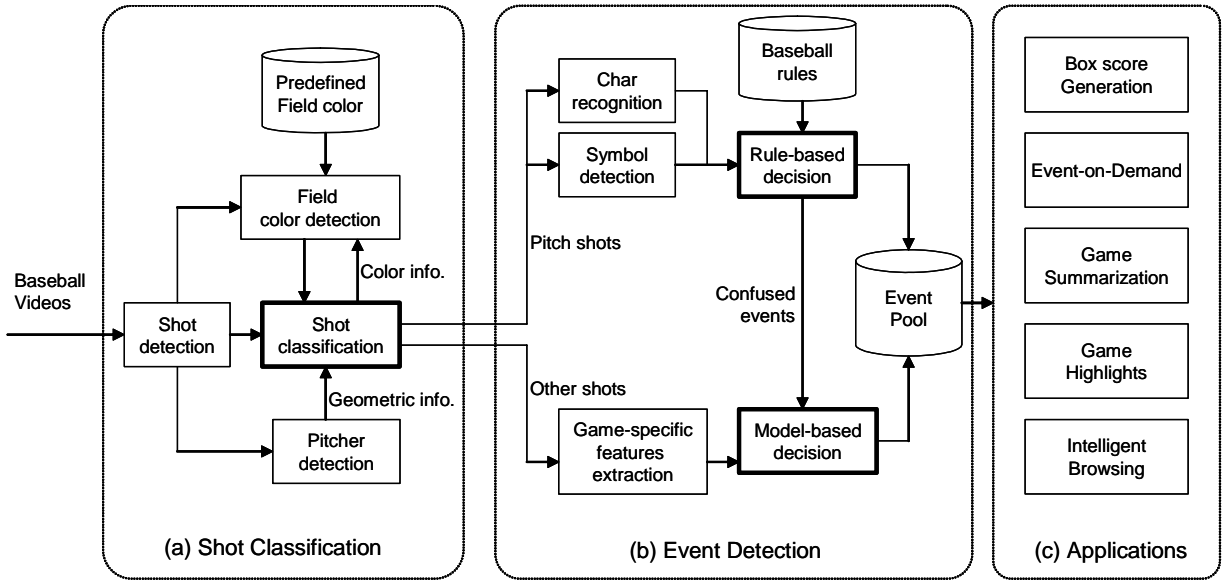
Figure 1. Examples of game progress.

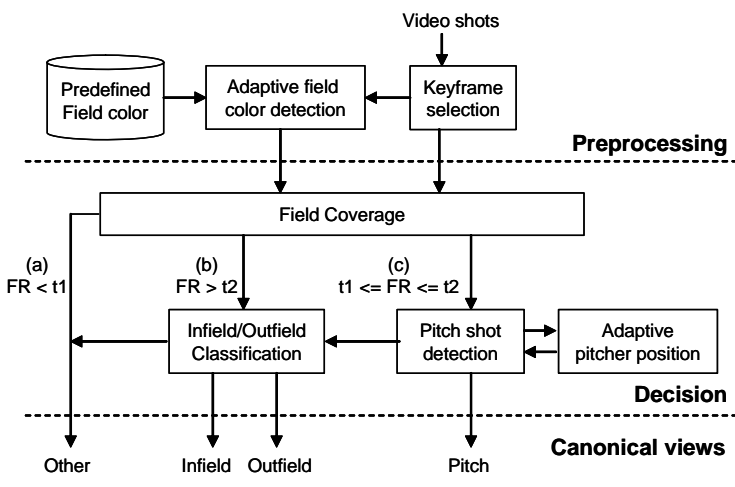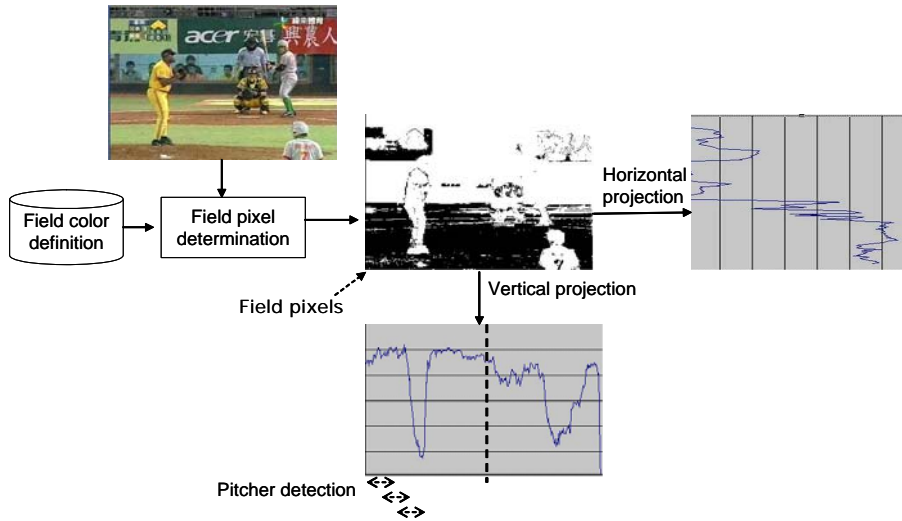**Figure 2. System framework of explicit event detection and its applications.**
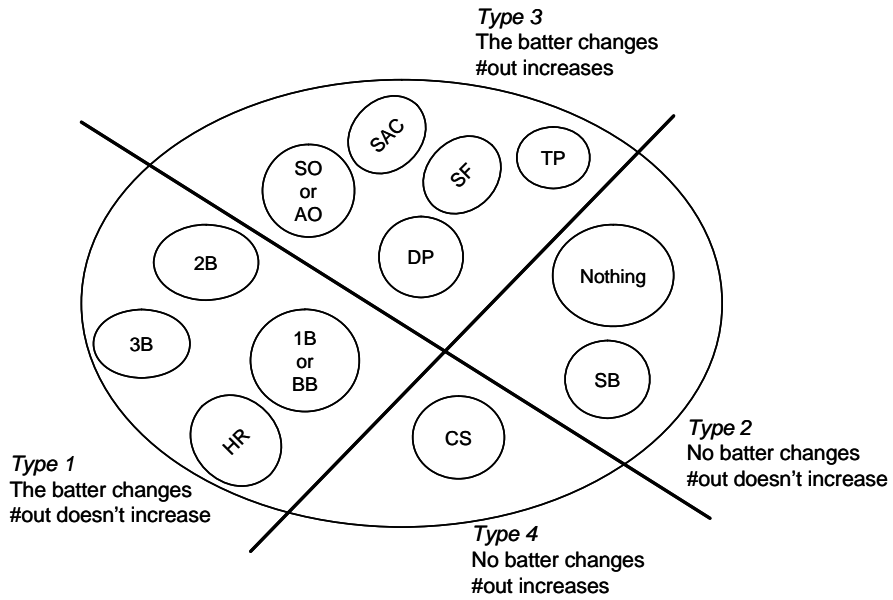

Figure 3. Diagram of shot classification.

Figure 4. Pitch shot detection by field pixel profiles and pitcher detection.

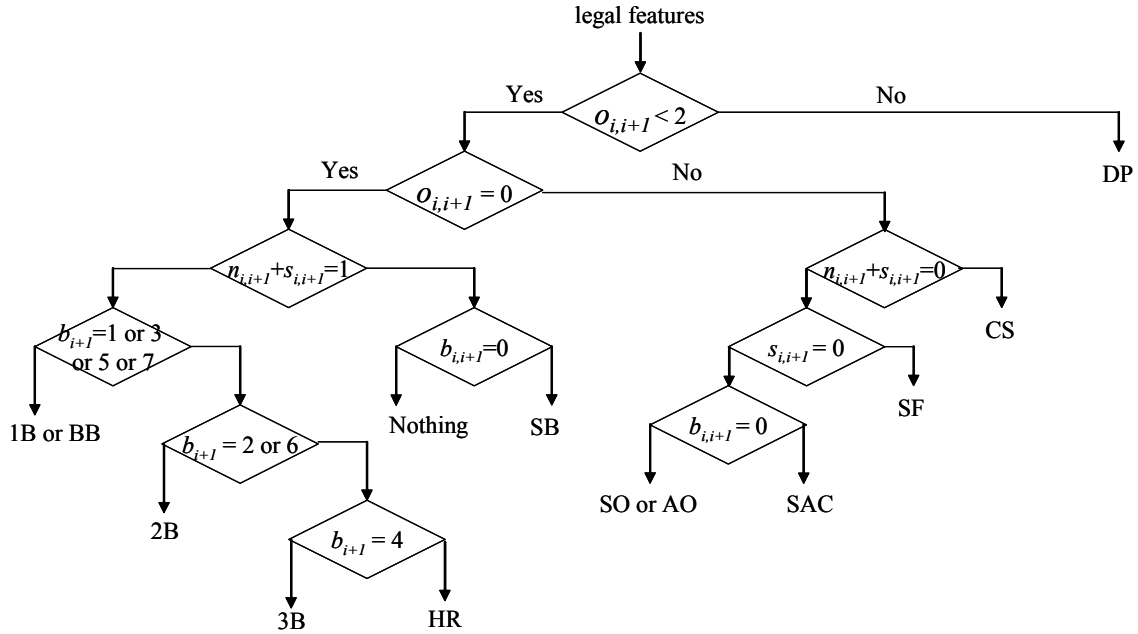**Figure 5. Taxonomy of baseball events.**

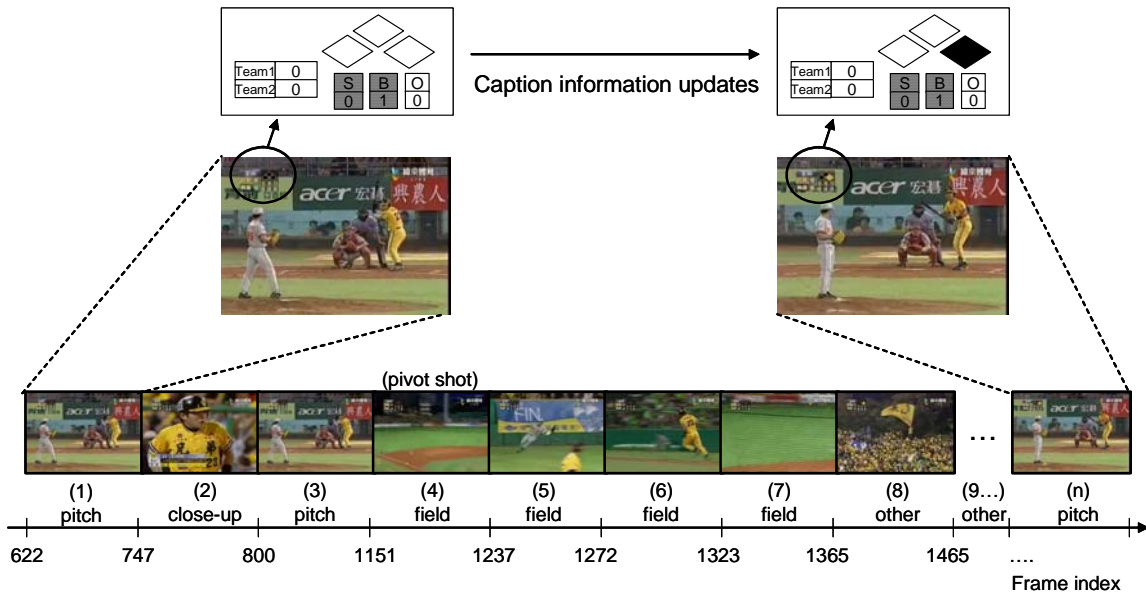**Figure 6. Event detection process on decision tree.**

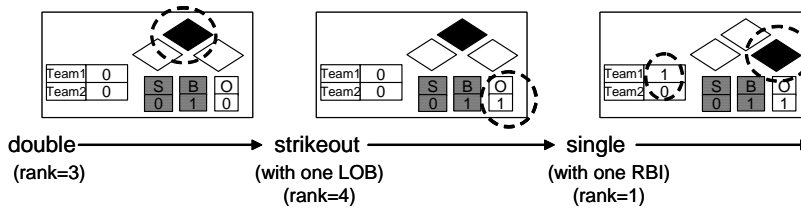**Figure 7. An example of shot context feature extraction.**

Figure 8. A chain of events that result in scoring.

Table 1. Physical meanings of different base-occupation situations.

b_i	n_i	Physical meaning
0	0	No base is occupied.
1	1	Only the first base is occupied.
2	1	Only the second base is occupied.
3	2	Both the first and the second bases are occupied.
4	1	Only the third base is occupied.
5	2	Both the first and the third bases are occupied.
6	2	Both the second and the third bases are occupied.
7	3	All bases are occupied.

Table 2. Confused event in baseball games

Confused events	Information changes between two consecutive pitch shots
Single and walk (base on ball, hit by pitch, or intentional walk)	The first base is occupied and no out increases.
Strikeout and field out	The number of out increases one and the number of occupied bases and score don't change.
Stealing, wild pitch, passed ball, and balk	No out increases and the runner advances to the next base.
Caught stealing and pickoff	The number of out increases one and the number of occupied bases decreases.

Table 3. Detection results of six common events.

Game		HB	Double	Home Run	O	Sacrifice	Double Play
CPBL1	Prc.	1 (15/15)	1 (6/6)	1 (2/2)	1 (35/35)	1 (4/4)	1 (3/3)
	Rec.	1 (15/15)	1 (6/6)	1 (2/2)	0.95 (35/37)	1 (4/4)	1 (3/3)
CPBL2	Prc.	1 (15/15)	1 (3/3)	1 (2/2)	1 (34/34)	1 (3/3)	0.75 (3/4)
	Rec.	0.83 (15/18)	1 (3/3)	1 (2/2)	0.89 (34/38)	1 (3/3)	1 (3/3)
CPBL3	Prc.	1 (17/17)	1 (3/3)		0.98 (43/44)	1 (1/1)	1 (2/2)
	Rec.	0.89 (17/19)	1 (3/3)		0.91 (43/47)	1 (1/1)	1 (2/2)
MLB	Prc.	1 (15/15)	1 (3/3)		1 (25/25)	0.6 (3/5)	0.75 (3/4)
	Rec.	0.79 (15/19)	1 (3/3)		0.81 (25/31)	1 (3/3)	0.75 (3/4)
Total	Prc.	1 (65/65)	1(15/15)	1 (4/4)	0.99(137/138)	0.85(11/13)	0.85(11/13)
	Rec.	0.92(65/71)	1(15/15)	1 (4/4)	0.90(137/153)	1(11/11)	0.92(11/12)

Table 4. Classification results of confused events.

Game		Single	Walk	Strikeout	Field out
CPBL1	Prc.	0.83 (10/12)	0.67 (2/3)	0.45 (5/11)	0.96 (23/24)
	Rec.	0.91 (10/11)	0.5 (2/4)	0.83 (5/6)	0.74 (23/31)
CPBL2	Prc.	1 (12/12)	1 (3/3)	0.8 (4/5)	0.93 (27/29)
	Rec.	0.8 (12/15)	1 (3/3)	0.44 (4/9)	0.93 (27/29)
CPBL3	Prc.	0.88 (7/8)	0.56 (5/9)	0.55 (11/20)	0.92 (22/24)
	Rec.	0.54 (7/13)	0.83 (5/6)	0.73 (11/15)	0.69 (22/32)
MLB	Prc.	1 (4/4)	0.73 (8/11)	0.3 (3/10)	1 (15/15)
	Rec.	0.36 (4/11)	1 (8/8)	0.33 (3/9)	0.68 (15/22)
Total	Prc.	0.92(33/36)	0.69(18/26)	0.5(23/46)	0.95(87/92)
	Rec.	0.66 (33/50)	0.86(18/21)	0.59(23/39)	0.76(87/114)

Table 5. Lengths of summaries at different levels.

Games	Length
Original game of 'Lions vs. Whale' (2005/6/14)	3 hours 33 minutes
Level-1 summary	4 minutes 10 seconds
Level-2 summary	23 minutes 15 seconds
Level-3 summary	51 minutes 32 seconds
Original game of 'Bulls vs. Lions' (2005/4/8)	3 hours 14 minutes
Level-1 summary	2 minutes
Level-2 summary	20 minutes 12 seconds
Level-3 summary	49 minutes 43 seconds

Table 6. Performances of different levels of summaries.

Lions vs. Whales (2005/6/14)					
Inning	**1**	**2**	**3**	**4**	**5**
Man-made summary	5	12	0	3	1
(1) Automatic summary (with context)	4/4	12/12	0	3/3	1/1
(2) Automatic summary (without context)	4/4	8/8	0	3/3	1/1
Inning	**6**	**7**	**8**	**9**	**Total**
Man-made summary	0	4	3	5	33
(1) Automatic summary (with context)	0	4/4	3/6	4/5	31/35
(2) Automatic summary (without context)	0	4/6	3/4	5/6	28/32

Bulls vs. Lions (2005/4/8)					
Inning	**1**	**2**	**3**	**4**	**5**
Man-made summary	0	4	4	6	0
(3) Automatic summary (with context)	0	4/4	3/3	6/8	0/4
(4) Automatic summary (without context)	0/2	2/4	3/3	5/7	0/2
Inning	**6**	**7**	**8**	**9**	**Total**
Man-made summary	7	0	5	4	30
(3) Automatic summary (with context)	4/4	0	5/5	3/3	25/31
(4) Automatic summary (without context)	2/2	0/1	2/2	3/3	17/25

Precision of (1) = 31/35 = 0.886, Recall of (1) = 31/33 = 0.939
Precision of (2) = 28/32 = 0.875, Recall of (2) = 28/33 = 0.848
Precision of (3) = 25/31 = 0.806, Recall of (3) = 25/30 = 0.833
Precision of (4) = 17/25 = 0.680, Recall of (4) = 17/30 = 0.567

Table 7. The selected events in 'Lions vs. Bears.'

Lions vs. Bears (2005/4/2)	
Inning	**Selected events**
Top 2nd	sacrifice fly (RBI=1)
Bottom 5th	hit by pitch
Top 6th	Walk double sacrifice fly (RBI=1)
Top 6th	sacrifice bunt single (RBI=1) double (RBI=1)
Bottom 8th	Single steal walk

Table 8. The selected events in 'Bulls vs. Lions.'

Bulls vs. Lions (2005/4/8)	
Inning	**Selected events**
Top 4th	home run (RBI=1)
	single
	home run (RBI=2)
Bottom 4th	sacrifice fly (RBI=1)
Bottom 6th	home run (RBI=2)
	field out (good defense play)
Bottom 8th	Walk
	Field out (good defense play)
Bottom 9th	Single
	sacrifice bunt
	sacrifice bunt

Table 9. The evaluation results of highlights from two games.

Game highlights	**Average mean opinion score**
Lions vs. Bears 7-min highlight (11 events)	3.35
Lions vs. Bears 5-min highlight (8 events)	3.43
Bulls vs. Lions 7-min highlight (11 events)	3.67
Bulls vs. Lions 5-min highlight (8 events)	3.87