Development of Realistic Applications Based on Explicit Event Detection in Broadcasting Baseball Videos

Wei-Ta Chu
Dept. of Computer Science and Information Engineering
National Taiwan University, Taipei, Taiwan
wtchu@cmlab.csie.ntu.edu.tw

Ja-Ling Wu
Dept. of Computer Science and Information Engineering
Graduate Institute of Networking and Multimedia
National Taiwan University, Taipei, Taiwan
wjl@csie.ntu.edu.tw

Abstract

This paper presents a framework that explicitly detects events in broadcasting baseball videos and facilitates the development of various extended applications. Three phases are included: reliable shot classification, explicit event detection, and elaborate applications. In the shot classification stage, color and geometric information are utilized to classify 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. Based on explicit event detection, many accurate and practical applications such as box score generation and game summarization can be built. The evaluation results show the effectiveness of the proposed methods and demonstrate some insights about bridging semantic gaps for sports videos.

1. Introduction

Sports video analysis has attracted much attention due to its potential commercial benefits and advanced digital home applications. Sports games that follow different rules and broadcasting characteristics draw different issues in video analysis. In terms of competition types, sports games can be distinguished into team sports (two teams face each other), competitions (more than two players compete), and individual competitions (athletes participate individually one after the other) [1]. In terms of game taxonomy, the most popular sports such as soccer [2-6], football [7], and baseball [8-12], are widely studied. Techniques that exploit audiovisual features or domain knowledge are proposed to tackle various problems in different sports. To the end of providing efficient media access or attractive entertainment functionalities, applications of scene classification [11,12], event detection[5,9,10,11,15], highlight extraction [1,7,8], or game summarization [3,13,14] are developed.

In this paper, we concentrate our work on broadcasting baseball videos and establish a system that materializes explicit event detection and practical applications. We emphasize the importance of explicit event detection and exploit game-specific features to scrutinize baseball videos.

Recently, three types of studies are applied to baseball videos: shot classification, highlight extraction, and event detection. In studies of shot classification, shots are classified as pitch, infield, outfield, audience, or close-up views [11,12]. Based on aural/visual characteristics, game highlights are automatically detected and game summarization could be generated [8]. These two kinds of techniques facilitate automatic game content analysis, whereas the reported results that just include rough game summarization are still far from practical usage. To a baseball fan, it's more attractive to see 'what really happened in the game' rather than 'rough game summarization'. Event detection techniques are, therefore, devised to analyze game videos at finer granularity.

Zhang and Chang [9] proposed an event detection method based on caption information. However, they only focused on detecting the last pitch and scoring. Han et al. [10] developed a baseball digest system based on maximum entropy method and detected seven baseball events. However, the detection performance is not very promising in the reported results.

According to the terms of Major League Baseball [16], more than ten common events lead the game progress. We emphasize the importance of explicit event detection and propose a framework that detects (almost) all possible events in broadcasting baseball videos. The results of explicit event detection facilitate the development of more attractive applications and provide a good foundation for more reliable game summarization.

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 based on visual features. 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 concluding remarks are given in Section 6.

2. SYSTEM FRAMEWORK

An important observation in baseball videos is that all meaningful events occur between two consecutive pitch shots. Thus the status changes within this duration pose important clues to indicate what happened and facilitate explicit event detection. 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.

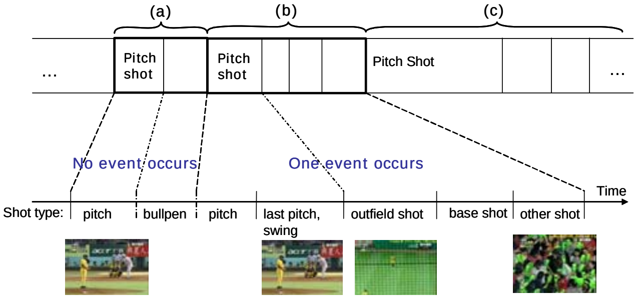


Figure 1. Examples of game progress.

In broadcasting baseball videos, the superimposed caption immediately reflects the game status. After an effective event, outs, scores, and/or base-occupation situation would change according to the actions of players. Information changes on the caption, therefore, provide significant clues about event type and the results caused by events. In this work, we transform official baseball rules into a decision tree structure and efficiently bridge the gap between audiovisual features and semantic events. As we elaborately exploit the inherent characteristics of broadcasting baseball videos, only caption information and shot transition information are needed to explicitly detect events. Accurate and efficient event detection is therefore able to be achieved.

Figure 2 shows the system framework, which includes three stages of processes: shot classification, event detection, and extended applications. In the shot classification stage, the baseball videos are first divided into shots, where color and geometric features are then extracted. An adaptive field color detection module is developed to dynamically adjust the field color range for different broadcasting situations. For the same reason, a reliable pitcher detection process provides geometric information to the classification module in detecting pitch views. Finally, video shots are classified into pitch, infield, outfield, or other views.

The event detection stage includes rule-based decision and model-based decision methods. As the essential idea described above, we specially tackle the caption information in pitch shots. The information changes between two pitch shots are computed. According to baseball rules, the rule-based decision module infers what happened within this duration. Then a model-based decision module that characterizes shot transition information is further developed to explicitly handle some confused events. After this process, various events are explicitly uncovered.

On the basis of explicit event detection, we can develop more practical and accurate applications. This system automatically generates box score and provides event-on-demand services. According to the consequence of different events, we further design summarization and highlight selection algorithms to produce game abstraction that better matches fans' need and expectation.

3. SHOT CLASSIFICATION

Several canonical views provide rough representation of games and are important clues to facilitate further analyses. We adaptively determine the color range of field and develop a color-based shot classification module.

3.1 Adaptive Field Color Determination

Color range of field may significantly change at different stadiums, in different TV channels, or at different times. Fortunately, field region dominantly occupies the bottom part of video frames. 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 every twenty minutes to analyze a 3-minute video clip and determine the latest field color range.

All procedures of shot classification are performed in HLS color space. For each color channel, pixel color is temporally collected to construct an integrated histogram that indicates the dominant color range within this duration. We check the 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.

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 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. Only the frames with enough suspected field pixels are taken into account. The newly determined color range then update the default field color definition. In our implementation, the initial settings of field color are: $0.05 < \text{hue} < 0.33$ and $0.2 < \text{saturation} < 0.327$, where the value of each color channel is normalized into the range [0,1].

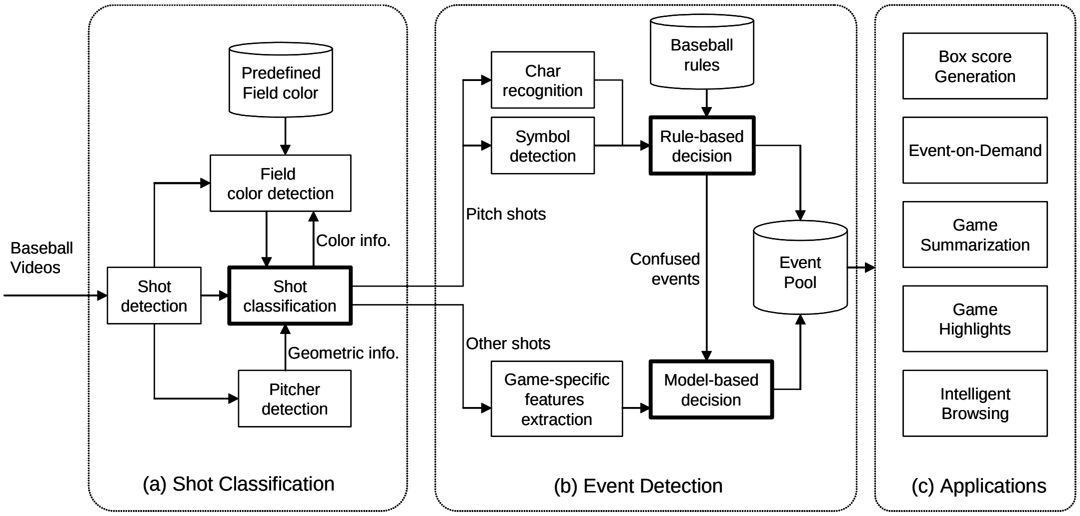


Figure 2. System framework of explicit event detection and its applications.

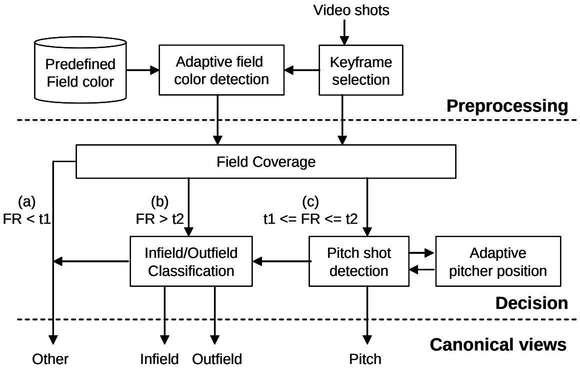


Figure 3. Diagram of shot classification.

3.2 Procedures of Shot Classification

Figure 3 shows the process of a color-based shot classification module. Color ranges of the field, including grass and soil, are adaptively determined by a field color determination module. With field color definition, we compute the ratio of field area to the keyframe, which is selected from each video shot. Two thresholds, *t1* and *t2* (*t1* < *t2*), are defined for classifying shots. The steps of classification are:

1) If the field ratio (FR) is less than 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 threshold *t2*, the corresponding shot is like the field. In general, the outfield view often contains audience or stadium artifacts and displays higher texture content. Therefore, an edge-based infield/outfield classification module is further developed to distinguish between infield and outfield views.
3) If the field ratio is between two thresholds, the corresponding shot is first verified by a pitch shot detection module. We detect field pixels with the help of field color definition. The spatial layout of field pixels is then checked by projecting them in horizontal and vertical directions. If this keyframe is a pitch view, the field pixels should concentrate only on the bottom part of horizontal profile. On the other hand, because the pitcher is always in the left part of a pitch shot, we can find a valley in the vertical profile.
   If the frame still doesn't conform to the definition of pitch view, it is further verified by infield/outfield classification module, which finally classifies it as infield, outfield, or other views.

In our system, thresholds *t1* and *t2* are empirically 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 fixed thresholds for field ratio. Through the processes described above, the keyframe of each video shot is classified as pitch, infield, outfield, or other view.

4. EVENT DETECTION

In this work, we elaborately exploit baseball rules to perform fine-granularity event analysis and bridge the semantic gap between low-level features and baseball events. For example, if no base is occupied in the *i*-th shot, and the score increases by one but still no base is occupied in the (*i*+*1*)-th shot, we can infer that a solo home run occurs. This simple idea effectively facilitates explicit event detection and precise video analysis.

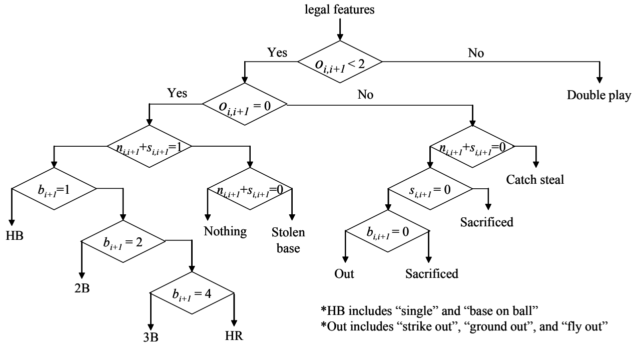
4.1 Rule-based Event Detection

Each effective baseball event leads to changes of caption information, such as 'home run' increases the score, 'strikeout' increases out, and 'single' and 'walk' change the base-occupation situation. Therefore, we can simply check the information changes on the caption and accomplish efficient implementation for event detection.

Caption information is often displayed as text (such as number of score) or symbol (such as number of out and base occupation). We perform character recognition and intensity-based detection to extraction caption information. For the duration between two consecutive pitch shots, the changes of out, score, and base-occupation situation are jointly considered in event detection. They are concatenated as feature vectors to characterize baseball events.

Conceptually, feature vectors derived from all possible baseball events construct a feature set. Given a feature vector, we can view the process of event identification as classifying it into a subset, which represents one baseball event. In this work, necessary condition rules derived from baseball regulations are applied. The necessary condition rules indicate that a specific event is necessary for some information changes on caption. For example, a double in the i-th pitch shot is necessary for the occupied second base in the ($i+1$)-th pitch shot. If there is no occupied base in the i-th pitch shot, the batter hits out the ball and reaches the second base, and the caption in the ($i+1$)-th pitch shot shows that the second base (and only this base) is occupied. If there are some occupied bases in the i-th pitch shot, the batter hits out the ball and reaches the second base, one (the second) or two (the second and the third) bases would be occupied in the ($i+1$)-th pitch shot. By considering the necessary condition of various events, the event identification process can be conceptually modeled as a decision tree, and events are determined after tracing the tree (c.f. Figure 4).

Thirteen events are considered in this work: single (1B), double (2B), triple (3B), home run (HR), stolen base (SB), caught stealing (CS), fly out (AO), strikeout (SO), base on ball (Walk, BB), sacrifice bunt (SAC), sacrifice fly (SF), double play (DP), and triple play (TP).



*HB includes "single" and "base on ball"
*Out includes "strike out", "ground out", and "fly out"

Figure 4. Event detection process on decision tree.

The rule-based process effectively detects most events by employing information changes on caption. However, event pairs such as 'single vs. walk' and 'strikeout vs. field out' lead to the same information changes and cannot be explicitly discriminated by simply using rules. To make the event detection process more explicitly, we develop a model-based approach that adopts contextual shot information and elaborate the detection results.

4.2 Model-based Event Detection

Contextual information of shot transition and its temporal duration often provide clues for us to identify events. We consider shot context features and develop classifiers to discriminate events that are implicitly hidden after rule-based event detection.

According to the observation of broadcasting style and baseball rules, we propose the following features to describe the characteristics of shot transition, temporal variations, and motion information. Note that these features are extracted within the duration from the end of previous effective event to current pitch shot.

- *ConsecutivePF*: indicating whether a field view displayed immediately after the last pitch view. The first field shot right after the last pitch shot 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 shots before the pivot shot. In general, more pitch shots occur before the pivot shot in the events of 'walk' and 'strikeout.'
- *DiffPitchField*: indicating the time difference between the last pitch shot and the pivot shot.
- *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.
- *Motion*: indicating the motion magnitude of the pivot shot. When the ball is hit out, the camera tracks the ball or the fielder and brings about higher motion.

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 other 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. After rule-based decision, the sequences decided as 'single or walk' candidates are further discriminated by the classifier. The same process is applied to detect 'field out' or 'strikeout'. In this work, k is empirically set as 8 for classification accuracy and efficiency. Details of event detection processes please refer to our previous work [17].

4.3 Results of Event Detection

The evaluation data are from games of Chinese Professional Baseball Leagues (CPBL) [18] and Major League Baseball [16]. Five broadcasting games with total length twelve hours are used. They are recorded directly from TV, and the commercials are not intentionally filtered out. Because the proposed framework only takes into account the caption information in pitch shots and shot transitions (pitch-field pairs), the commercials that are often classified as 'other' shots don't significantly 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 the game.

Table 1 shows the detection results of six common events. The term 'HB' denotes 'single' or 'walk', and 'O' denotes 'strikeout' or 'fly out'. The value n1/n2 denotes that n1 events are detected by the proposed method, and n2 indicates the ground truth. The detection results are very promising, while only one false alarm in 'HB' and 'SAC', and one miss in 'O' in CPBL1. The detection performances of CPBL3 and MLB2 are slightly worse than the other cases because of worse shot classification or character recognition accuracy deriving from poorer video quality. Note that although only common events are shown in Table 1, other rare events could also be correctly detected by the proposed method. For example, the only 'triple' events in CPBL2 and MLB1 and the only 'catch out' event in CPBL3 are correctly detected.

Table 2 shows the classification results of discriminating confused events, i.e. 'HB' and 'O' events in Table 1. The classification results are also very satisfactory in CPBL1 and CPBL2 games, while they present larger variations in other games. The shot transition patterns often differ slightly in different TV channels and different situations. However, through the simple modeling method, the proposed framework achieves satisfactory performance without being drastically affected by game variations. A system demo about explicit event detection can be seen at http://www.cmlab.csie.ntu.edu.tw/~wtchu/baseball/.

Table 1. Detection results of six common events.

Game	HB	2B	HR	O	SAC	DP
CPBL1	17/16	5/5	2/2	32/33	5/4	3/3
CPBL2	16/17	3/3	2/2	33/35	4/4	4/3
CPBL3	11/12	3/3	1/1	27/33	3/4	0/0
MLB1	11/11	1/1	0/0	20/21	2/2	1/1
MLB2	17/19	3/3	0/0	25/31	6/4	2/4

Table 2. Classification results of confused events.

Game	Single	Walk	Strikeout	Field out
CPBL1	14/12	3/4	8/6	24/27
CPBL2	13/14	3/3	7/9	26/26
CPBL3	6/9	4/3	11/15	16/18
MLB1	4/7	6/3	6/4	14/17
MLB2	7/11	10/8	9/9	16/22

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 explicit event detection, many practical applications can be developed for efficient access, information filtering, or entertainment purposes.

5.1 Automatic Generation of Box Score

By integrating the results of explicit event detection with a given lineup, we can gather the performance of each player in a game and automatically generate box score. Users can click events of one specific player and view the corresponding video clips immediately. We believe that event-based access is at adequate granularity and is a practical example of multimedia indexing and cross-modal retrieval.

5.2 Automatic Game Summarization

In the literature, many studies of sports video analysis are conducted for game summarization or highlight extraction. Among them, works of summarization focus on reserving as much as information in games, while works of highlight extraction focus on collecting segments that greatly excite users. We develop both summarization module that deals with 'informativeness' and highlight extraction module that deals with 'enjoyability' based on the results of explicit event detection.

5.2.1 Significance Degree of Events

To jointly maintain informativeness and effectiveness of game summary, we give different significance degrees to different types of events. Babaguchi et al. [7] 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 approaches and modify the definition of significance especially for baseball videos:

- 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.

- Rank 2: hits with RBIs. Hits with RBIs, no matter they are single, double, or home run, change score of a team and indirectly affect the result of the game. They also indicate the effectiveness of hits.
- Rank 3: hits without RBI and walk. Although no score is obtained, the number of hits is concerned with a player's batting average.
- Rank 4: outs with LOBs. They show that the batter fails to help teammates score. Larger LOB indicates more negative influence when a play makes an out.
- Rank 5: outs without LOB. Normal outs generally cover more than half of cases and give the least significance.

5.2.2 Selection of Summarization

We provide various summaries that have different lengths and information coverage. The most compact condensed game is generated by concatenating rank-1 events, while a richer condensed game 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 to maintain the completeness of summary. For example, in Figure 5, 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 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 lonely (no other events with ranks ≦ 3), it should be ignored because fragmentary hits don't cause effective results.

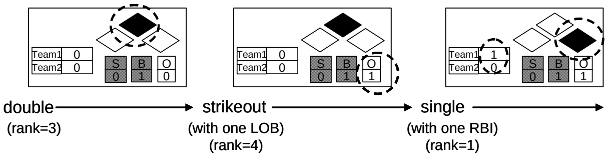


Figure 5. A chain of events that result in scoring.

On the basis of ideas described above, three levels of game summaries are automatically generated by the following methods.

- Level 1: Only the events with rank 1 and rank 2 are collected.
- 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 i.
    1) 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.
    2) If $r_{i+1} < 3$ and $r_{i+2} = 5$, just pick the i-th and (i+1)-th events as the summary.
    3) If $r_{i+1} = 4$ and $r_{i+2} = 5$, ignore all the i-th, (i+1)-th, and (i+2)-th events.
- Level 3: All events with ranks 1~5 are collected. This level of summary contains the most complete content of a game, while eliminating irrelevant clips.

Note that the temporal relationships between events should be maintained, because a condensed game formed by disordered events is meaningless. Therefore, different ranks of events may be interlaced in the final summary. The 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.

5.2.3 Evaluation of Summarization

To evaluate the effectiveness of the proposed summarization method, we compare level-2 summary with man-made condensed games taken from a sport TV channel [19]. Although the selected events may not be the same from different reporters or different channels, they can be viewed as good references for evaluation.

Table 3 shows the summarization performance of two different games. The value n2/n1 in automatic summary denotes that n1 events are collected by the proposed process, and among them, n2 events are in man-made summary. From Table 3, the precisions of two level-2 summaries are 25/31=0.806 and 31/35=0.886, respectively. The corresponding recalls are 25/30=0.833 and 31/33=0.939.

Table 3. Performances of different levels of summaries.

Bulls vs. Lions (2005/4/8)	
Man-made summary	30
Level 1 sum.	3/3
Level 2 sum.	25/31
Lions vs. Whales (2005/6/14)	
Man-made summary	33
Level 1 sum.	7/7
Level 2 sum.	31/35

5.3 Automatic Highlight Generation

Another attractive application is the game highlight extraction. Generation of highlights poses different concerns from summarization to maintain entertaining functionalities within short time duration. It is evident that effective events such as state change events or hits with

RBIs should be highlighted. In addition, good defense play such as diving catch or catch steal should be highlighted, although they just cause a normal field out. In highlight extraction, we integrate the impacts of event ranks, occurrence time, and audio energy dynamics to generate game highlight that well retains ‘enjoyability’ of a game.

5.3.1 Significance Degree of Events

- Rank-based Significance

We slightly modify the definition of event rank. Double play, triple play, and catch steal events are categorized as rank-2 events to cover important defense. The rank-based significance degree S_r ($0 \le S_r \le 1$) of the i-th event E_i is quantitatively defined as

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

where r_i ($1 \le r_i \le 5$) denotes the rank of the i-th event, and α ($0 \le \alpha \le 1$) is the parameter controlling the weight of event rank.

- Time-based Significance

The events occurring at the latter or final stage of games are usually more attractive to users. The time-based significance S_t ($0 \le S_t \le 1$) is defined as

$$S_t(E_i) = 1 - \frac{N - I(E_i)}{N} \cdot \beta, \tag{2}$$

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

- Audio-based Significance

The anchorperson often comments excitedly and the audience cheers loudly when a beautiful play or an important hit occurs. We extract audio energy and analyze its dynamics over time to show how it reacts to each event.

The audio-based significance S_a ($0 \le S_a \le 1$) is defined as

$$S_a(E_i) = 1 - \frac{5 - D_i}{5} \cdot \gamma, \tag{3}$$

where D_i represents the maximum energy difference in an event duration and is quantized into one of five levels. γ ($0 \le \gamma \le 1$) is the parameter controlling the weight of audio cues.

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

$$S(E_i) = S_r(E_i) \cdot S_t(E_i) \cdot S_a(E_i). \tag{4}$$

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

5.3.2 Highlight Selection Algorithm

To select the most highlighted events, the significance degrees of events and the obtained highlight length are considered. As regards the event length, we just extract an event from the last pitch to the first pitch of next event. This method facilitates just reserving the most significant parts and provides efficient event presentation.

We implement a greedy approach to select highlighted events. By considering the time limitation given by the user and event context, the 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($T, E$)
$A \leftarrow \varnothing$
sort $E$ into nonincreasing order by significance degrees
for each $e_i \in E$
    do if length of $(A \cup \{e_i\}) < T$
        then $A \leftarrow A \cup \{e_i\}$
            SMOOTH($A$)
return $A$
```

In the case of three adjacent events A-B-C, the SMOOTH process selects the event B if both events A and C selected as highlight. Finally, the selected events are sorted by the occurrence time to maintain temporal coherence.

5.3.3 Evaluation of Highlight

Due to lack of ground truth for evaluating game highlight, we invited 24 persons, including 21 males and 3 females, to perform subjective experiments based on highlights extracted from two games. We impose two assumptions on the subjects: 1) none of the subjects saw these games before. We cannot expect every subject affords to spend more than six hours to see two baseball games. 2) The subjects judge the selected highlighted events based on the events themselves rather than their preference on specific teams or specific players.

The experiment scenario is set to be event-based. Because of the assumption 1, 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. This evaluation somehow represents the ‘accuracy’ of the proposed highlight selection method. Because an event’s significance sometimes depends on the effectiveness of the following events, we present multiple events together if they are in the same half inning. After the presentation of one half inning, the subjects give one opinion score (higher score means better result) to each selected event to indicate whether it’s a highlight part.

Table 4 shows the subjective evaluation results of highlights with different lengths. Eleven events and eight events are selected to construct 7-min and 5-min highlights, respectively. In Table 4, 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', two home runs occurred and two teams have a keen competition. On the other hand, the team 'Lions' dominates the game in the 'Lions vs. Bears' sequence, and the game presents flat content. Therefore, the events selected in 'Bulls vs. Lions' often excite the subjects more and get higher scores.

Table 4. The evaluation results of highlights from two games.

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

6. DISCUSSION AND CONCLUSION

We have presented a complete strategy to perform explicit event detection and develop practical applications for broadcasting baseball videos. Color and geometric information is utilized to achieve shot classification. 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. 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. Finally, on the basis of explicit event detection, automatic game summarization and highlight selection are implemented to preserve 'informativeness' and 'enjoyability' within short duration. The proposed approaches automate baseball video analysis and facilitate bridging semantic gaps for sports videos. In the future, commentator's speech may provide hints for events not decided from caption information, such as player substitution. Furthermore, various applications with different purposes and services can be developed.

7. ACKNOWLEDGE

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

8. REFERENCES

[1] M. Bertini, A. Del Bimbo, and W. Nunziati, Highlights Modeling and Detection in Sports Videos. Pattern Analysis and Applications, 2005.

[2] 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 CSVT, 14, 5, 2004, 634-643.

[3] A. Ekin, A.M. Tekalp, R. and Mehrota, Automatic Soccer Video Analysis and Summarization. IEEE Transactions on Image Processing, 12, 7, 2003, 796-807.

[4] 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,* 2003, 11-20.

[5] 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 Intl. Workshop on MIR,* 2004, 127-134.

[6] 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, 25, 7, 2004, 767-775.

[7] N. Babaguchi, Y. Kawai, T. Ogura, and T. Kitahashi, Personalized Abstraction of Broadcasted American Football Video by Highlight Selection. IEEE Transactions on Multimedia, 6, 4, 2004, 575-586.

[8] Y. Rui, A. Gupta, and A. Acero, Automatically Extracting Highlights for TV Baseball Programs. In *Proc. of ACM Multimedia,* 2000, 105-115.

[9] D. Zhang and S.-F. Chang, Event Detection in Baseball Video Using Superimposed Caption Information. In *Proc. of ACM Multimedia,* 2002, 315-318.

[10] M. Han, W. Hua, W. Xu, and Y. Gong, An Integrated Baseball Digest System Using Maximum Entropy Method. In *Proc. of ACM Multimedia,* 2002, 347-350.

[11] D. Zhong and S.-F. Chang, Real-Time View Recognition and Event Detection for Sports Video. Journal of Visual Comm. and Image Representation, 15, 2004, 330-347.

[12] W. Hua, W. Han, and Y. Gong, Baseball Scene Classification Using Multimedia Features. In *Proc. of ICME,* 2002, 821-824.

[13] 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, 15, 2004, 393-424.

[14] D. Tjondronegoro, Y.-P. P. Chen, and B. Pham, Integrating Highlights for More Complete Sports Video Summarization. IEEE Multimedia, Oct.-Dec., 2004, 22-37.

[15] 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,* 1, 2003, 25-28.

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

[17] W.-T. Chu and J.-L. Wu, Integration of Rule-based and Model-based Methods for Broadcasting Baseball Videos, In *Proc. of ICME,* 2005.

[18] Chinese Prof. Baseball League, http://www.cpbl.com.tw

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