Automatic Parsing of Sports Videos with Grammars

Fei Wang
Institute of Computing Technology
Chinese Academy of Sciences
Beijing, P.R. China, 10080

Kevin J. Lü*
Brunel University
Uxbridge, U. K. UB8 3PH

Jingtao Li
Institute of Computing Technology
Chinese Academy of Sciences
Beijing, P.R. China, 10080

* Corresponding author. Tel:+44-1895203122; Fax:+44 1895 203 149;
E-mail address: kevin.lu@brunel.ac.uk (Kevin J. Lü)

Abstract: Motivated by the analogies between languages and sports videos, we introduce a novel approach for video parsing with grammars. It utilizes compiler techniques for integrating both semantic annotation and syntactic analysis to generate a semantic index of events and a table of content for a given sports video. The video sequence is first segmented and annotated by event detection with domain knowledge. A grammar-based parser is then used to identify the structure of the video content. Meanwhile, facilities for error handling are introduced which are particularly useful when the results of automatic parsing need to be adjusted. As a case study, we have developed a system for video parsing in the particular domain of TV diving programs. Experimental results indicate the proposed approach is effective.

Keywords: video parsing**,** sports videos, semantic annotation, and syntactic analysis

1 Introduction

Digital videos have become more and more popular and the amount of digital video data has been growing significantly. As a result, efficient processing of digital videos has become crucially important for many applications. Most of current video systems are still unable to provide the equivalent functions, like “table of contents” or “index” which are available for a textbook, or for locating required information. Because manual video annotation is time-consuming, costly and sometime can be a painful process, various issues of content-based video analysis and retrieval have been intensively investigated recently [1, 2]. The key problem that needs to be resolved is that of automatically parsing videos, in order to extract meaningful composition elements and structure, and to construct semantic indexes.

This study is concerned with the automatic parsing of sports videos. As a great favorite of a large audience over the world, sports videos represent an important application domain. Usually, a sports game has a long period, but only part of it may need to be reviewed. For example, an exciting segment from a one-hour diving competition may only last a few seconds – from jumping from the springboard to entering the pool. It’s discouraging to watch such a video by frequently using the time-consuming operations of “fast-forward” and “rewind”. Thus, automatic parsing of sports videos is highly valued by users, for it not only helps them to save time but also gives them with the pleasing feeling of control over content that they watch [3]. Moreover, efficient tools are also useful to professional users, such as coaches and athletes, who often need them in their training sessions.

The task of sports video parsing is similar to creating an index and a table of contents for a textbook, which encompasses two subtasks:

1) extracting index entries based on semantic annotation; and

2) constructing a comprehensive structure hierarchy based on content structural analysis.

Most related previous work on sports videos has its focus on semantic annotation with shot classification [4, 5], highlight extraction [6, 7], and event detection [5, 8-10]. A video shot is referred to as an unbroken sequence of frames recorded from a single camera, and usually it is the basic unit in video processing. Based on domain-specific feature extraction, such as color, edge, and motion, Neural Networks [4] and Support Vector Machines [5] were used to classify shots into predefined categories. In order to extract the most interesting segments or highlights of a sports video, the method based audio-track analysis [6] and the method by modeling user's excitement [7] were proposed separately. However, the lack of exact semantics is the main drawback in those approaches. The end users will almost always like to interact with high-level events when accessing or retrieval sports video segments, such as a serve in tennis, or a goal in soccer. In [8], several high-level events in tennis videos were detected by reasoning under the count-line and player location information. In [9], they first determined candidate shots in which events are likely to take place by extracting keywords from closed caption streams, and then those candidates were matched and selected with example image sequences of each event. Both the rule-based approach [5] and the statistical-based approach [10] were used to infer high-level events by employing context constraints of sports domain knowledge. Although significant progress has been made on automatic semantic annotation, it is still hard to obtain sufficient accuracy when handing the vast amount of video content in real environment.

Structural analysis is another important issue, which has been mentioned in the literature [11, 12].

However, their approaches are restricted to segmenting fundamental units such as serve and pitch in tennis, play and break in soccer. In [13], a general-purpose approach was proposed which does not require an explicit domain model. It adopts the time-constraint clustering algorithm to construct a three-layer structure, i.e., shot, group and scene. However, such an unvarying structure representation is not suitable for sports videos owing to the lack of the ability to model various game structures. Thus, none of the existing work is capable of recognizing the hierarchical game structures of sports videos.

The aim of this paper is to introduce a novel approach to integrate both semantic and structural analysis for sports videos parsing with grammars. Different from other systems in the literature, we suggest that sports videos could be treated as languages, where the sport video parsing system is analogous to a compiler. Our system consists of three procedural steps: basic unit segmentation, semantic annotation and syntax analysis. Firstly, the raw video stream is segmented into basic units, which are equivalent to words in a language. Although there exist different units, such as shots, sub-shots, or other predefined segments, we treat the shot as the basic unit due to it's ubiquity in sports video analysis. Secondly, each basic unit is annotated during semantic analysis. This step detects semantic events and assigns tokens indicating these events to the basic units. Finally, we utilize context-free grammars to represent the content inter-structures of sports videos, because the grammars provide a convenient means for encoding the external rules into the application domain with a parse tree. Based on the grammars, we employ the syntax analysis to identify a hierarchical composition of the video content. Meanwhile, with the use of the grammars, our system would be able to identify misinterpreted shots and to detect errors since automatic analysis based on low-level features cannot provide 100% accuracy. To our best knowledge, this study is the first attempt to integrate semantic annotation and syntactic analysis for

parsing sports videos. Experimental results show that our system is effective and easy to use. Although we only demonstrate parsing diving competition videos as a case study in this paper, the framework can also be applied to other sports videos.

The rest of the paper is organized as follows. Section 2 presents our approach to modeling sport videos. This is followed by a framework for automatic parsing of TV diving programs in Section 3. Experimental results are reported in Section 4. Section 5 concludes the paper.

2 Modeling Sports Videos

In many types of sports broadcasting, one can have the following two interesting observations. First, each sports game can be represented in a tree structure. For example, a tennis game is divided first into sets, then games and serves. A diving game contains several rounds, and there are some plays in each round as shown in Figure 1. In order to facilitate user access, efficient techniques need to be developed to recognize the tree structure from raw video data.

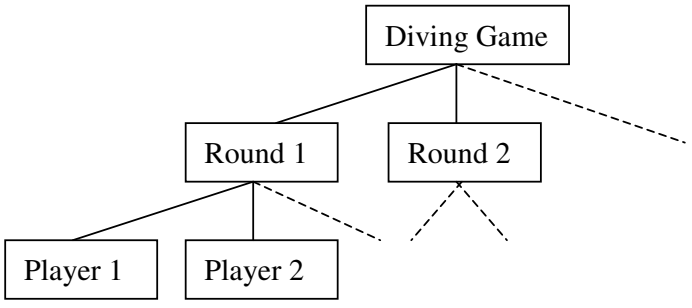

Figure 1. Hierarchical structure of a diving competition

Second, there is a number of repetitive domain-specific events in sports videos, which are meaningful and significant to users. These events can be classified into three groups: *replay events, state events* and *target events* (see Figure 2). In sports videos, interesting events are often replayed in slow motion immediately after they occur. We call the replay segments as *replay events*. *State events* occur when the game state is changed, such as score, introduction of players before a play begins. Because they typically indicate the beginning and the end of structural units, *state events* are highly correlated with the game structure. Following these, we consider *target events*, which represents specific objects and their motions in a game, such as shots in soccer games or dives in diving competitions.

(a) (b) (c)

Figure 2. Examples of events in sports videos:

(a) replay events, (b) state events, and (c) target events.

Due to a wide variety of video content, it is almost impossible to provide a versatile method of event detection, which is able to bridge the gap between the low-level features and the high-level semantics.

Thus, we have devoted a great deal of attention to the application context. Based on our observations from sports videos, we reveal that

- *replay events* typically are sandwiched between specific shot transitions;
- *state events* are usually accompanied with superimposed captions, which are overlapped on the video in the production process to provide information about the situation of the game; and
- in *target events*, motion introduced by objects and cameras is much active, often synchronized with the audience's cheers and game-specific sounds (e. g. water hits in diving games, baseball hits in baseball games).

Based on the above observations, sports video parsing is similar to language processing which is based on dictionaries and grammars. In the scope of sports videos, the dictionary that we use to annotate shots is a set of domain-specific events, and the grammar is a set of rules represented in the form of the tree structure.

3 A Framework for Parsing Sports Videos

In this section, we first introduce a framework which provides the system overview and then discuss the related algorithms for semantic and structural analysis. To show the merit of our approach, we develop a system for parsing TV diving programs as a cast study.

3.1 Overview

The purpose of this framework is to parse a sport video to construct a semantic index and a table of

contents based on events. Through the use of the index and the table of contents, users will be able to position specific video contents which they are looking for. The proposed system, which is a compiler-like, is composed of three phases: shot detection, semantic annotation and syntactic analysis. Figure 3. shows the flowchart of the framework.

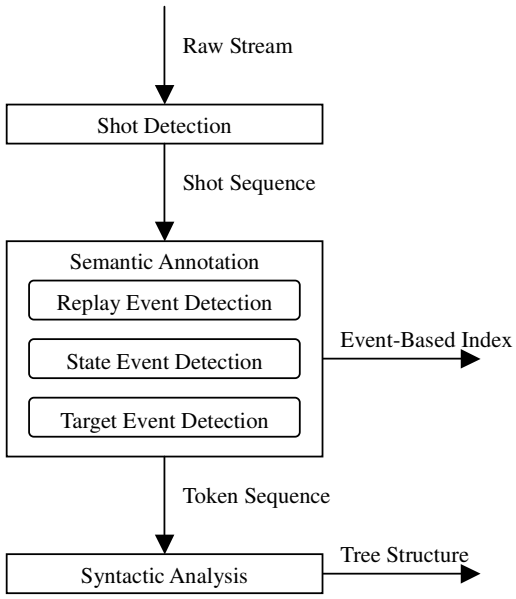

Figure 3. Architecture of the system

First, the raw stream is segmented into a sequence of shots by using automatic shot boundary detection techniques. A number of algorithms have been proposed for this purpose and we implemented a histogram-based approach, which achieves a satisfactory performance for both abrupt and gradual shot transitions [14].

Second, shots are recognized as tokens based on semantic event detection. Each event is associated with a token. For example, the token “d” represents the dive event in diving competitions. After an event is detected, every shot in the event is annotated with the token, which can be used as an event-based index.

We regard an event as a stochastic temporal process. Different event detection methods can be integrated into our framework. Currently, in our system for TV diving programs, three domain-specific approaches are proposed including replay event detection, state event detection and target event detection.

Finally, we use the sequence of tokens to construct a tree structure. Every sport game has its own rules that are the base that the structure that the game needs to follow. Prior to parsing, the syntax of the sports game is described by a context-free grammar. Then we exploit compiler techniques to design a parser. Meanwhile, error detection and recovery procedures are implemented in the syntactic analysis phase.

3.2 Semantic Annotation

In nature, the semantic annotation is the process in which each shot is classified by predefined event models. As stated in Section 3, the events are divided into three categories: *replay events*, *state events*, and *target events*. In this section, we introduce different algorithms for recognizing each of them in a diving video.

3.2.1 Replay Event Detection

Replay events are sandwiched between special shot transitions, which usually contain logos with special editing effects. We have developed a straightforward algorithm for automatic detection of the *replay event*, which includes the following steps:

1. The pixel-wise intensity distance is measured between the frames in shot boundaries and the

example logo images at the region where logo images typically appears. If the distance is below an empirically chosen threshold, then the special shot transition is detected.

2. If the interval between two special transitions is in the range of a possible duration for a replay, a *replay event* is recognized and all shots between the transitions are annotated with the *replay event*.

In a diving competition, the *replay event* is the replay segment focusing on the dive of the competition. The above method has utilized the fact that the duration of a *replay event* is far shorter than the interval between *replay events*. The example logo images can be viewed as a template of the logo. How to obtain it is a crucial issue. Ideally, the logo template may be learned from example streams. At the current stage, we only randomly select some example frames as the template from the concerned video stream. Based on the best match, the distance is calculated.

3.2.2 State Event Detection

State events are normally accompanied by superimposed captions providing important information about the status of the game. In a diving competition, there are three kinds of *state events* including "ready", "score", and "round end". "Ready" is the event when the player gets ready on the platform or springboard. The superimposed text includes player's name, rank, DD (Degree of Difficulty), etc. After that, the player dives into the pool. When the player climbs out the pool, the text box of the score appears which is defined as the event "score". The event "round end" refers to the end of a round associated with a scoreboard. Superimposed text in different *state events* has different layout and keywords. In our system, the three types of *state events* can be detected.

A number of algorithms to extract caption text from video have been published in recent years, while most of which have to fully decompress the video sequence before extracting text regions. Because digital videos are usually stored in compressed forms for efficient storage and transmission (such as MPEG), we could save on system resources and time needed for decompression, by manipulating features directly in the compressed domains. We implemented a method similar to the one used in [15], which can locate candidate caption text directly in the DCT compressed domain for MPEG video.

After the text (existing in the form of "text blocks", i.e., a rectangle box that covers a line of text) in each frame is detected and obtained by automatic text detection, we measure the similarity between the frame and the example image of the *state event*.

Let $F = \{f_1,\ldots,f_n\}$ and $G = \{g_1,\ldots,g_m\}$ denote the text blocks in the frame and the example image respectively. $|f|$ or $|g|$ is the number of pixels in each text block, and $f \cap g$ is the set of joint pixels in f and g. In the matching, the similarity is given by

$$s(F,G) = \frac{\sum_{f\in F}\sum_{g\in G}\tau(f,g)\,|\,f \cap g\,|}{\max(\sum_{f\in F}|\,f\,|,\sum_{g\in G}|\,g\,|)} \tag{1}$$

where

$$\tau(f,g) = \begin{cases} 1, & \text{if } \min(|f \cap g|/|f|,|f \cap g|/|g|) \geq 0.7 \\ 0, & \text{else} \end{cases}$$

If the similarity is beyond a threshold, the frame would be matched with the *state event*. We count the matched frames in a shot, and assign the shot with the token of the *state event* that has the most matched frames. If few frames are matched, the shot doesn't belong to any *state event*.

3.2.3 Target Event Detection

As discussed in Section 2, most of the *target events* can be well characterized by motion. In a diving competition, we are pursuing the "dive" as the *target event*. In fact, it is one shot, in which an athlete dives into the pool from the platform or springboard. The camera focuses on the athlete, and at the moment of diving, there is a dominant downward camera motion. Therefore, the camera motion can be used as a critical cue.

In the current version of our system, we use a camera motion detector to recognize and model events. There are several methods available for camera motion analysis, such as optical flow computation. For estimating the camera motion between two successive frames, we first calculate motion vectors from block-based motion compensation, and then the vectors are counted to infer the camera motion. Because the camera usually puts athletes at the center of the view in a diving competition, for example, we don't calculate the motion vectors near the center of frames, which could reduce the computational cost as well as the false estimation caused by the front objects (i.e. the athletes). This approach may not be very accurate, but fast speed of computation can be achieved. When detecting a diving event that is about 1 or 2 seconds, a sliding window of width w is used. For a 25-fps video, w is set to be 25 frames. The dive event is declared only if more than half of the w frames in the current window are found to be tilt down.

3.3 Syntactic Analysis

To introduce the syntactic analysis for sports video parsing is essential for three reasons. First, by use it, we can efficiently construct the tree structure based on compiler techniques. Second, by describing the knowledge about the game structures with grammars, we can separate the domain knowledge from the parsing process. Thus, the system is more flexible and can be easily extended. Third, a new facility of error handling can be introduced. It also helps users to locate errors in the results of automatic parsing, which could make the system more friendly and usable.

Table 1. Tokens in a diving game

Token	**Category**	**Semantics**
r	replay event	replay segment
b	state event	be ready for a dive
s	state event	score
e	state event	end of round
d	target event	Dive
u	undefined shot	undefined shot

Once the sports video is annotated with the tokens by the event detection (see Table 1), we need to identify the structure by the syntactic analysis. The approach used in the syntactic analysis is similar to a language compiler, which builds a parse tree from the input sequence according to the grammar. Here, the stream of tokens produced by the semantic annotation is parsed, and then based on the grammar description to construct the table of contents for a specific game.

We use context-free grammars to describe the syntax of sports games. For example, the tree structure of a diving competition (as shown in Figure 1) can be expressed as following:

$S \rightarrow R \mid RS$

$R \rightarrow Pe \mid PR$

$P \rightarrow bdrs$

where S is the start symbol, R means a round which consists of P – the play of each diver. We ignore undefined shots as "blanks" between events. If several shots in succession are annotated by the same token, the first one is fed to the parser while the left one is skipped. By elimination of left factoring, the grammar is translated to the LL grammar, and then a predictive parser is used to construct a parse tree.

Because the tokens recognized by automatic semantic annotation may be inaccurate and the actual video content may not be confirmed with the grammar. How to respond to errors is an another task of the syntactic analysis. None of the existing video parsing systems have addressed this problem. Many of them have pointed out that video content analysis should not be operated as a fully automatic process, which should request manual adjustments. In our system, we introduce a new facility for error handling. It is particularly useful when the results of automatic parsing need to be validated manually. The objectives of our error handling facility includes: (1) to report the error occurrence timely and precisely; (2) to recover from an error for later analysis; and (3) it should not seriously reduce the speed of normal processing. If an error occurs long before it is detected, it is difficult to identify precisely what is the nature of the error. For the viable-prefix property, (i.e. an error is detected at the moment that the prefix of the input cannot be a prefix of any string of the language), the LL method that we used can detect an error as it happens. To recover errors, in general, several strategies have been widely accepted and used,

including panic model, phase level, error production, and global correction. The panic model is used for the simplicity and efficiency in our system, where the parser discards the input symbol until a designated set of synchronized tokens are found (delimiters as “e”).

When an error is reported, the syntactic analysis can be easily intervened and controlled by users. Based on the compiler-like parser architecture, the video content analysis in our system is an iterative process, in which each loop of the iteration includes automatic parsing, error reporting, and manual adjustment. Thus, it provides an interactive environment for video parsing.

4 Experimental Results

Our system has been implemented on a Pentium IV 1.8GHz PC using Java language with Java Media Framework API under Windows 2000. Figure 4 shows a screen dump of our system. After the user opens a diving video and uses the automatic parsing tool, a table of contents and an index will be returned to the tabbed pane in the left-hand side. A table of contents leads to a hierarchical composition of shots, in which each node is also represented with a label and a key frame. An index entry is a textual label with a key frame from the event. Users can use the representative frames and the text annotations to search the global content of the video, or to view an interesting part by clicking the representative frame, without the need of doing tedious “fast-forward” and “rewind”. In addition to facilitate user’s access to the content, the system also provides an interactive environment for content analysis. That is, users can check the error reports from the syntactic analysis and adjust the labels by additional operations, such as “Add”, “Remove”, and “Annotate”.

Figure 4. Video parsing for TV diving programs

Table 2. Data Set

	Length	Replay Event	State Event	Target Event	Total
Game A	0:46:54	40	84	40	164
Game B	0:44:13	40	85	40	165
Game C	1:09:24	60	125	60	245
Game D	1:26:43	72	150	72	294
Total	4:07:14	212	444	212	868

To assess and evaluate the system, we tested it by parsing four diving competition videos (see Table 2), with the digitization rate equal to 25 frames/sec in MPEG format of 352×288 frame resolution. The videos come from different competitions and stadiums, including Game A (3m Synchronized Diving

Men), Game B (10m Synchronized Diving Women), Game C (3m Springboard Diving Women), and Game D (10m Platform Diving Men). The ground truth is labeled manually.

The experiments carry two objectives. The first is to evaluate the event detection based on the semantic annotation. The second is to evaluate the performance of the syntactic analysis.

4.1 Result of Semantic Annotation

Two different levels of evaluation for the semantic annotation were performed: shot-level and event-level. The shot-level evaluation checks if the annotation of every shot is correct. The accuracy is defined as the ratio of the number of shots being correctly annotated over the number of shots. The test results are shown in Table 3. For the four video clips, the accuracy rates are all above 90%, which demonstrates the effectiveness of our approach.

Table 3. Result of semantic annotation on shot level

	Accuracy
A	327/356=92%
B	420/448=94%
C	645/673=96%
D	770/850=91%
Total	2162/2327=93%

The event-level evaluation is measured by the *precision* and *recall* rates for each type of events:

$$\text{precesion} = \frac{\text{number of correctly detected events}}{\text{number of detected events}}$$

$$\text{recall} = \frac{\text{number of correctly detected events}}{\text{number of events}}$$

An event is referred as *correctly detected* if any shot of the event is annotated correctly. From Table 4, our system achieves better performance on *replay events* and *state events* than on *target events*. Comparing contents of these three types of event, we found that *target events* are generally different from *state events* and *reply events.* We believe the reason lies in the large motion variation in the video shots. To enhance the performance, more effective features and more powerful statistical models are required.

Table 4. Result of semantic annotation on event level

	Replay Event		State Event		Target Event	
	Precision	Recall	Precision	Recall	Precision	Recall
A	40/40=100%	40/40=100%	78/78=100%	78/84=93%	30/43=70%	30/40=75%
B	40/40=100%	40/40=100%	78/78=100%	78/85=92%	25/34=74%	25/40=63%
C	60/60=100%	60/60=100%	114/116=99%	114/125=91%	58/71=82%	58/60=97%
D	71/73=97%	71/72=99%	118/119=99%	118/150=79%	58/84=69%	58/72=81%
Total	99%	100%	99%	87%	74%	81%

4.2 Result of Syntactic Analysis

In our experiments on diving competitions, high-level structure units beyond shots include play and round. A play is defined as the segment from the event “ready” to the event “score”. A round is the interval between the events “round end”. Unlike [13], in which the structure that most people agreed with was used as the ground truth of the experiments, our definition and evaluation are more objective. From the results in Table 5, it is observed that the proposed approach never made a false detection, but tended to miss some high-level units. This is because in the grammar-based syntactic analysis, a high-level unit

is defined in terms of not only events occurring but also the relations between them. Namely, an event may be missed because some events associated with it are detected wrong. A more powerful strategy of error recovery may resolve this problem. However, we would like to point out that this problem isn't so important in an interactive environment for video parsing, because the switch between automatic parsing and manual editing is convenient and smooth. After the parser detects an error, it may stop to wait for the user's modification and resubmission.

Table 5. High-level structure construction results

	Shots	Play			Round		
		Detected	Miss	False	Detected	Miss	False
A	356	34	6	0	4	0	0
B	448	34	6	0	5	0	0
C	673	49	11	0	5	0	0
D	850	50	22	0	6	0	0
Total	2327	167	45	0	20	0	0

Table 6. Error report in the syntactic analysis

	Annotation Error	Reported Error	Missed Error
A	29	22	7
B	28	18	10
C	28	22	6
D	80	40	40
Total	165	102	63

In Table 6, we assess the ability of error detection in the syntactic analysis. The LL method is able to detect an error as soon as possible. However, it is always difficult to correct the error immediately

without manual interruption. In the pane mode of error recovery in our system, the parser recovers itself until a synchronizing token is found. Due to the interval before the parser gets recovered from an error and is ready for detecting the next error, some errors may be missed. In the current system 62% of errors are reported. Considering the simple strategy that we adopted, the results are very encouraging.

5 Conclusions

In this paper, we have proposed a novel framework for video parsing with grammars. Motivated by the analogies between languages and sport videos, we introduced integrated semantic and structural analysis for sports videos by using compiler principles. Video table of contents and indexes based on events provide users with a semantic way of finding the content in which they are interested. In addition, the grammars enables users to identify errors in the results of automatic parsing, which could make the system more friendly and usable. As a case study, a video parsing system for TV diving programs has been developed.

At present, we are extending this framework to other typical sports videos (i.e., volleyball, tennis, and basketball). The remaining problems are from two challenges: 1) to enhance the event detection, e.g., more audio-visual feature representations and machine learning techniques; 2) to extend the grammar-based parser to handle loose structure patterns like basketball and soccer, where stochastic grammars may be better.

6 Acknowledgements

This work was partly funded by State General Administration of Sports of P. R. China. We would like to thank the coaches and athletes in Chinese International Diving Team for their valuable opinions. Kevin J. Lü would like to show his appreciation to Wang Kuan Cheng Science Foundation for the funding to enable him to conduct his research.

7 References

[1] C.W. Ngo, H.J. Zhang, and T.C. Pone, "Recent Advances in Content Based Video Analysis," *International Journal of Image and Graphics*, December 2001.

[2] N. Dimitrova, H.J. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, "Applications of Video-Content Analysis and Retrieval," *IEEE Multimedia*, vol. 9, no. 4, 2002.

[3] F.C. Li, A. Gupta, E. Sanocki, L. He, and Y. Rui, "Browsing Digital Video," *Proc. ACM Conference on Human Factors in Computing Systems*, pp. 169-176, April 2000.

[4] J. Assfalg, M. Bertini, C. Colombo, and A.D. Bimbo, "Semantic Annotation of Sports Videos," *IEEE Multimedia*, vol. 9, no. 2, 2002.

[5] L.Y. Duan, M. Xu, T.S. Chua, Q. Tian, and C.S. Xu, "A Mid-level Representation Framework for Semantic Sports Video Analysis," *Proc. ACM Multimedia*, November 2003.

[6] Y. Rui, A. Gupta, and A. Acero, "Automatically Extracting Highlights for TV Baseball Programs," *Proc. ACM Multimedia*, pp. 105-115, October 2000.

[7] A. Hanjalic, "Generic Approach to Highlights Extraction from a Sports Video," *Proc. IEEE International Conference on Image Processing*, September 2003.

[8] G. Sudhir, J.C.M. Lee, A.K. Jain, "Automatic Classification of Tennis Video for High-Level Content-Based Retrieval," *Proc. IEEE International Workshop on Content-Based Access of Image and Video Databases*, pp. 81-90, January 1998.

[9] N. Babaguchi, Y. Kawai, and T. Kitahashi, "Event Based Indexing of Broadcasted Sports Video by Intermodal Collaboration," *IEEE Trans. Multimedia*, vol. 4, no. 1, March 2002.

[10] G. Xu, Y.F. Ma, H.J. Zhang, and S.Q. Yang, "A HMM Based Semantic Analysis Framework for Sports Game Event Detection," *Proc. IEEE International Conference on Image Processing*, 2003.

[11] D. Zhong and S.F. Chang, "Structure Analysis of Sports Video Using Domain Models," *Proc. IEEE International Conference on Multimedia and Expo*, Aug 2001.

[12] L. Xie, S.F. Chang, A. Divakaran and H. Sun, "Structure Analysis of Soccer Video with Hidden Markov Models," *Proc. International Conference on Acoustic, Speech, and Signal Processing*, May 2002.

[13] Y. Rui, T.S. Huang, and S. Mehrotra, "Constructing Table-of-Content for Videos," *Multimedia Systems*, vol. 7, no. 5, 1999.

[14] R. Lienhart, "Comparison of Automatic Shot Boundary Detection Algorithm," *Proc. SPIE Storage and Retrieval for Image and Video Databases*, 1999.

[15] Y. Zhong, H.J. Zhang, and A. K. Jain, "Automatic Caption Localization in Compressed Video," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 22, no. 4, April 2002.