Constructing a Bowling Information System with Video Content Analysis

Wen Wen Hsieh and Arbee L.P. Chen
Department of Computer Science
Notional Tsing Hua University
Hsinchu, Taiwan 300, R.O.C
{wwhsieh, alpchen}@cs.nthu.edu.tw

ABSTRACT

In this paper, we present a design and implementation of a bowling information system. This system contains three types of bowling game information including the bowling video content information, the game-related information and the player information. The MPEG-7 Description Schemes are used to describe these types of information and the relationships among them. This information is obtained through an annotator by which manual conceptual feature annotation (for the player and game-related information) and automatic perceptual feature extraction (for the video content information) are integrated. Several interesting events in the video such as strikes and the important frames are determined by automatically analyzing the video content. With an interactive user interface, users' queries are transformed into XQuery to retrieve needed information about the bowling games to learn the skills of bowling. In addition to the implementation of the system, we also perform experiments to show the effectiveness of the automatic video content information extraction.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *retrieval models, selection process*. H.3.4 [**Information Storage and Retrieval**]: Systems and Software – *Question-answering (fact retrieval) systems*.

General Terms

Management, Design

Keywords

video analysis, video content extraction, bowling events, MPEG-7, video summarization

MMDB'03, november 7, 2003, new orleans, louisiana, usa.

1. INTRODUCTION

Information associated with videos can be categorized into three types [2]. The first one is the information which does not directly concern with the video content, such as the personal information of someone appearing in a video. The second one refers to the semantics of the video content, such as the score of a ballgame. The third one is concerned with the *perceptual features* of the video contents, for example, the moving object trajectory. The first and second types of information are also called *conceptual information*. A video retrieval system should provide sufficient information for the users to query and browse the video database.

Since users are more familiar with the semantic concept of a video, deriving semantics from perceptual features can provide a more friendly way for users to retrieve and browse video databases. The Multinet proposed in [4] uses the probabilistic models to map low-level features to high-level semantics. Sudhir et al. [7] analyze tennis videos and map the result to real events such as baseline-rallies. Another method [8] is designed to detect a complete set of the semantic events which happen in a soccer game. In [5], the results of analyzing both audio and visual features are used to detect the goal segments in a basketball game. In [1][6], similar ideas are used to detect the scenes in a video.

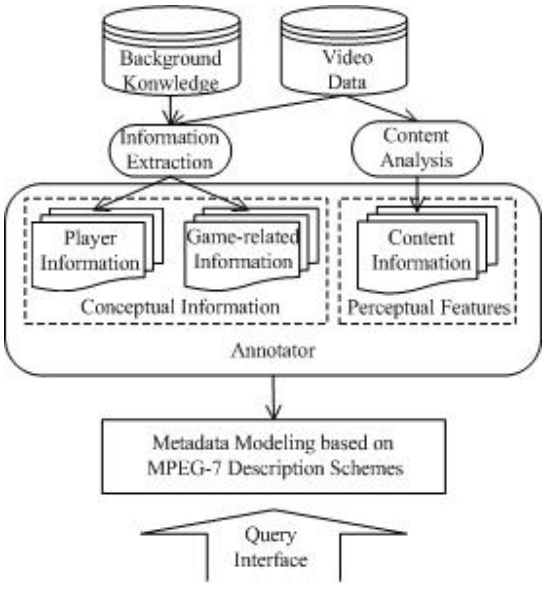

Figure 1. The framework of the bowling information system

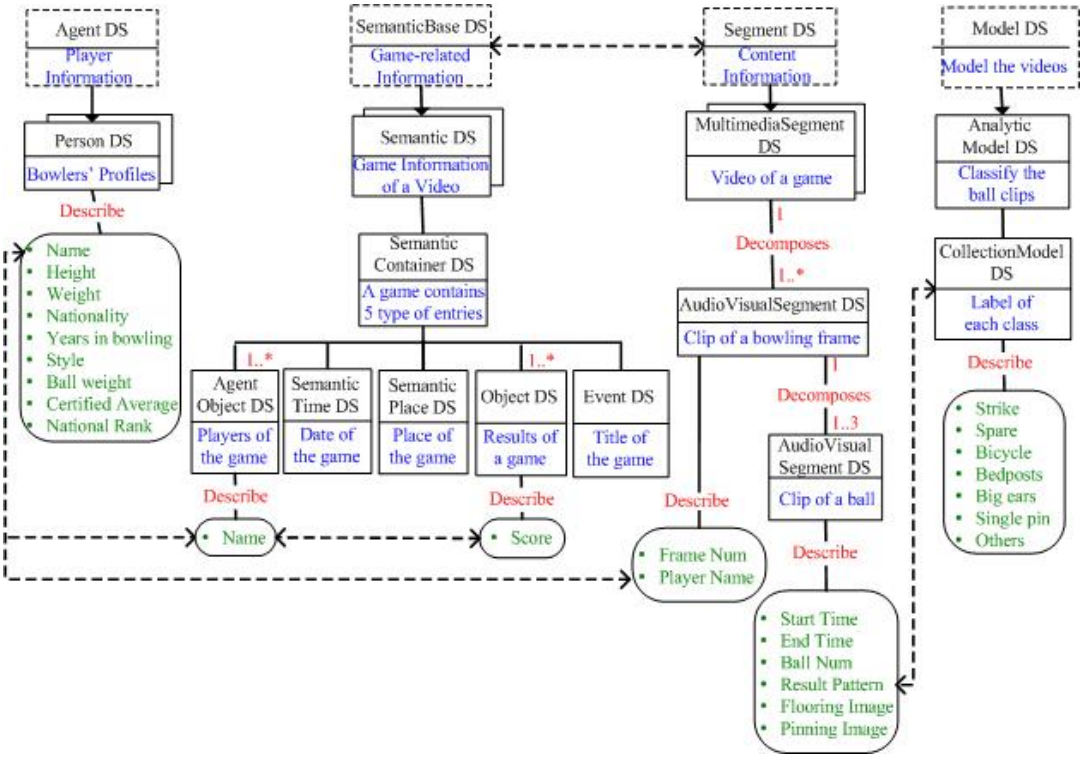

Figure 2. The system model based on MPEG-7

Moreover, after extracting the features, some specific training mechanisms are applied to obtain more accurate results. The differences between these work depend on the features and the training methods used.

In this paper, a bowling information system is designed and implemented, which contains the three types of information [2] to describe a bowling game information system. Based on the MPEG-7 Description Schemes [10][11], these three types of information and the relationships among them are described. Although the system proposed in [3] also uses MPEG-7 to manage and retrieve the soccer games, the player information and the game-related information are not considered. In our system, the most important frames such as the frame of the ball hitting the pins and the most interesting events such as a *strike* are useful for users to learn the skills of bowling. Since it is time-consuming to manually obtain these events and frames, we propose a method to analyze the visual and audio features to automatically extract this information. An annotator is designed to facilitate the annotation process which includes the automatic extraction of video content information. Finally, we perform experiments to show the effectiveness of the content analysis.

The rest of this paper is organized as follows. Section 2 presents the framework and the model of the bowling information system. Section 3 describes the video content analysis. Experimental results on the perceptual feature detection are shown in Section 4. Finally, Section 5 concludes this paper.

2. THE BOWLING INFORMATION SYSTEM

Figure 1 illustrates the framework of our bowling information system. The video content information, such as strike, is extracted through the content analysis procedure. In addition, some of the game-related information which contains descriptions associated with the game can be extracted from the captions of the video data while some other can be obtained from the background knowledge. Moreover, the player information such as the height and weight of a bowler is also considered and recorded as the bowler profile. All types of information are stored in XML file format because XML is the Data Definition Language (DDL) of the MPEG-7. Although some information can be automatically extracted, some of the information like the player information cannot be obtained from the video data directly. We design a semi-automatic annotator to collect all the information. This annotator is further described in section 3.

Our bowling information system is modeled based on the MPEG-7 Multimedia Description Schemes (DSs). Figure 2 shows the detailed metadata of the DSs. The dotted rectangles indicate the *abstract concepts*. The rectangles with solid lines indicate the concepts derived from the corresponding abstract concepts. They describe objects or events in bowling games. The rounded rectangles represent the attributes or features of the objects or events. The relationships between these elements are represented by the lines connecting these elements. The metadata are separated into four major parts. The dotted bi-directional links indicate the relationships between elements in different parts,

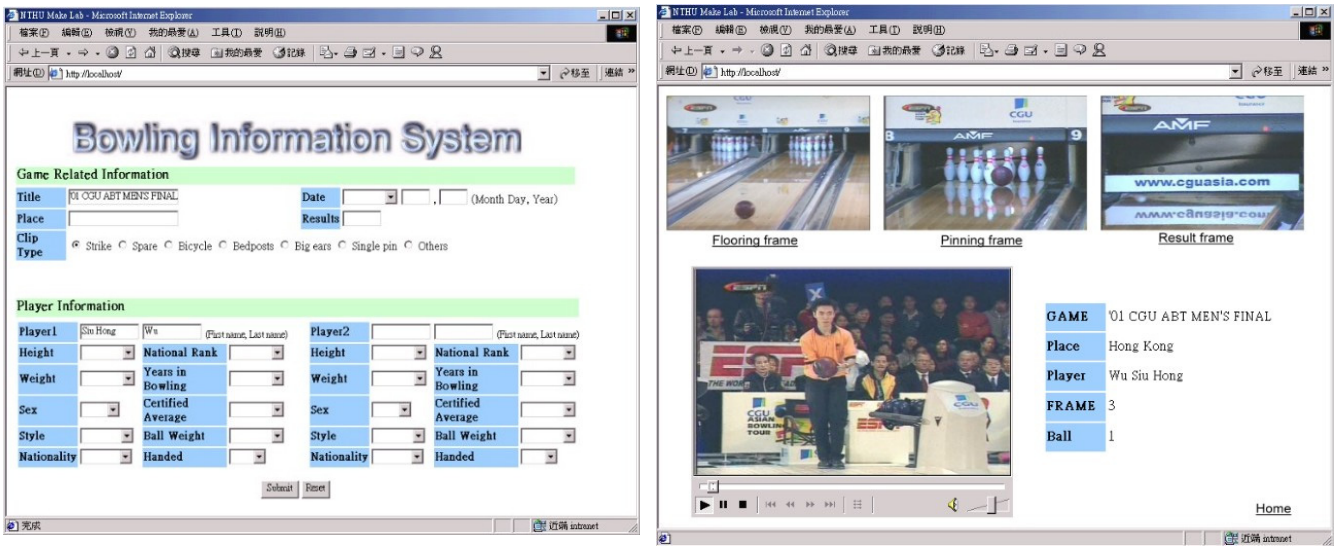

Figure 3. An example query and one of the results

which contain the same content. The four major parts are described as follows:

- **Player information**: It describes a bowler with his/her personal information including name, years in bowling, bowling style, certified average, etc.
- **Game-related information**: A bowling game is described by five semantic entities which are the title, the scores, the place, the date, and the bowlers of the game. Each bowler and the corresponding score are also specified.
- **Video content information**: The information is used to describe the structure of the video content. A video can be decomposed into a number of clips which represent *bowling frames*[1]. Each bowling frame has bowlers and identifier as the features. By the bowling rule, one bowling frame contains at most three *ball clip*s. A ball clip is a basic unit for the content analysis. It starts with the *address*[2] of a bowler and ends with the appearance of the sweep bar. By displaying a whole process of a bowler rolling a ball, a ball clip helps users to watch a bowler's entire movement, the ball track and the result caused by the ball. Some important information such as the result pattern (e.g., strike) is described in a ball clip.
- **Characterization of Videos**: Since the result of each ball is the most interesting part in a bowling game, the ball clips are collected and classified based on their results. Therefore, the users can retrieve a special kind of ball clips efficiently.

In the video content information, in addition to the result patterns, three most important frames are used to summarize a ball clip. They are:

- The *flooring frame*: it shows the ball hitting the floor. It captures the spatial relationship between the ball and the arrows marked on the floor. It can be checked to see whether a correct release point is achieved by the bowler.
- The *pinning frame*: it shows the ball hitting the pins. It can be checked to see whether a correct angle of the ball hitting the pins is achieved.
- The *result frame*: it shows the result of the ball. It is the last frame of a ball clip.

Users can interact with the bowling information system through an interface by specifying certain values on a part or across any of the four major parts. For example, users can retrieve the video of a certain game or all strike clips by a certain bowler, or question which blower is with the longest years in bowling. For the player information, the functions such as maximum and minimum are provided. Since the information is stored as XML files, each query posed in the query interface is transformed to XQuery for searching the matched data, and then the corresponding video clips are retrieved for the user. Figure 3 is an example query. There are seven clips retrieved and one of the retrieved clips is shown. The user can play the retrieved clip; moreover, three important frames are also displayed as a brief summary of the clip.

3. VIDEO CONTENT ANALYSIS

3.1 The Annotator

In this section, we describe the annotator to show how it works. The annotator shown in Figure 4 is designed as a wizard for a step-by-step annotation. Figure 4(a) shows the starting of the annotation process. In this step, a bowling games video is selected for annotating, and the game-related information about this game can be recorded. Figure 4(b) shows the annotation of the player information. The player information can be individually recorded.

On the contrary, a game cannot be successfully annotated unless the information about the players participating in the game has been recorded. After the game-related information and the player information have been annotated, we go to the next step to annotate the video content information.

Figure 4(c) shows the annotation of a ball clip. Since a ball clip is a unit for content analysis, to correctly segment all ball

[1] One of ten "innings" in a bowling game.

[2] The bowler's stance before beginning the approach.

Figure 4. The annotator

clips is important. We segment a video into ball clips by marking their start times and end times. Therefore, the users can browse particular ball clips with some interesting events. The functions Go, Slow and Fast control the operations of the video playing, and the function Get obtains the time points for a ball clip. After a ball clip is segmented, the information about the result can be extracted by analyzing the visual feature of the result frame, which is the last frame of the ball clip. The flooring frame and the pinning frame of this ball clip are also automatically determined by analyzing the audio features of the video. After the analysis, the two important frames pop in the window as Figure 4(d) shows. The frame number and the ball number are also automatically determined. In order to avoid the false determination, a fine-tune function is also provided.

3.2 Automatic Extraction of the Flooring/ Pinning Frames

The audio feature helps to decide the time points of both the flooring frame and the pinning frame since the sounds of a ball hitting the floor and a ball hitting the pins are different from the surrounding sounds. We analyze the audio data in the time domain and the frequency domain for precisely getting the time points. Since the audio signal is only stable and stationary in a very short time duration, it is necessary to use frame-based processing when extracting audio features. An *audio frame* is a collection of consecutive *samples*, and an audio is separated into a number of audio frames and each audio frame overlaps its previous one by a fixed number of samples.

In the time domain, the loudness which represents the energy of the audio is a feature used in the analysis and the root-mean-square (RMS) [9] is used to calculate the signal magnitude within each frame. Since the two hits usually make louder sounds after a duration which is close to silence, the time point of the two hits can be detected by the audio frame with a local maximum RMS value.

There are also commentators and the audience in the games who make noises. If the sounds of the two hits are not louder than the surrounding sounds in the audio data, the RMS method cannot distinguish them from others. Therefore, we also analyze the audio data in the frequency domain to help detect the two hits. Each audio frame is transformed into frequency domain by the fast Fourier transform to estimate its spectrum. The frequencies of a human's voice range from 85-Hz to 1.1-kHz. In addition, experiment results show the frequencies of the two hits range from 500-Hz to 1.5-kHz. Therefore, for the mean value of the magnitudes of the frequencies in each frame, the larger the value is, the higher possibility may it be the hit.

After the time points are chosen from both time domain and frequency domain, each of them is mapped to the corresponding video frame. The two hits can be captured and the flooring frame and the pinning frame can then be determined.

3.3 Automatic Detection of the Results

The visual feature is used to analyze the result frame to determine the result of the ball clip. There are four steps in processing.

1. Pre-processing: transform the result frame into an edge-based binary image and then reduce the noise effects. The transformation is needed because the edge-based binary image is more suitable for displaying the structure of the image. Moreover, the noise reduction can increase the accuracy of the analysis for an image.

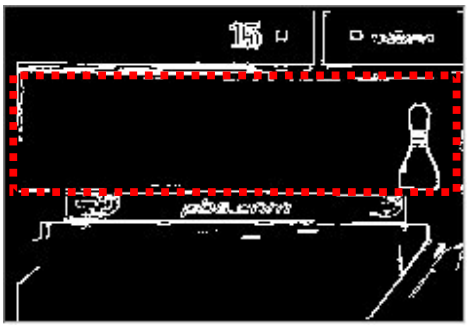

Figure 5 The ROI and the result frame after pre-processing

2. Determination of the *region of interest* (ROI): an ROI is a specific rectangular area in the result frame, and the dotted rectangle marked in Figure 5 is an example. The result of a ball clip can be determined by only analyzing the content of the ROI. Therefore, the top and the bottom boundaries of the ROI should be detected first. It can be constructed by the *horizontal projection* on the image. In an m × n image, the horizontal projection on raw $j, j=1,...,m$, is defined as

$$h(j) = \sum_{i=1}^{n} F(i,j) \qquad F(i,j) = \begin{cases} 1, & if \ \ p(i,j) = 255 \\ 0, & if \ \ p(i,j) = 0 \end{cases} \quad \cdots\cdots(1)$$

where $p(i, j)$ is the value of the pixel (i, j) in the edge-based image. The adjacent local maxima with the largest distance are regarded as the boundaries of the ROI.

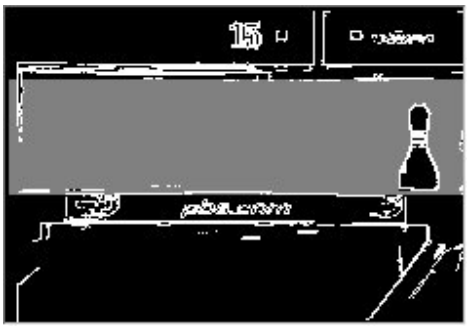

Figure 6. The ROI after the boundary- fill algorithm is applied

3. Detection of strikes and spares: by the *vertical projection* on the ROI, most cases of strikes and spares can be easily detected in this step. If there are no pins remaining on the pin-desk[3], the result of the vertical projection will have no peaks and the process terminates; otherwise, the process goes to next step. The criterion used here is to make sure that a strike detected in this step is a real strike. Some cases of a strike cannot be detected in this step due to noises, and such cases should be further processed in next step. The effectiveness of the filtering is discussed in section 4.

4. Detection of other interesting events: the final operation aims to detect interesting events by determining the positions of the pins remaining on the pin-deck. To reduce noises, a boundary-fill algorithm is utilized. The approach is to color the ROI except the regions of the remaining pins, and then the vertical projection on the ROI is used again. Figure 6 shows the result of applying the boundary-fill method to the image in Figure 5, and Figure 7 shows the result of the vertical projection of the image in Figure 6. Each pin can be identified by analyzing the result and the interesting events are detected based on calculating the corresponding position between the pins. The locations of the ten pins on the pin-deck in the tenpin bowling are shown in Figure 8.

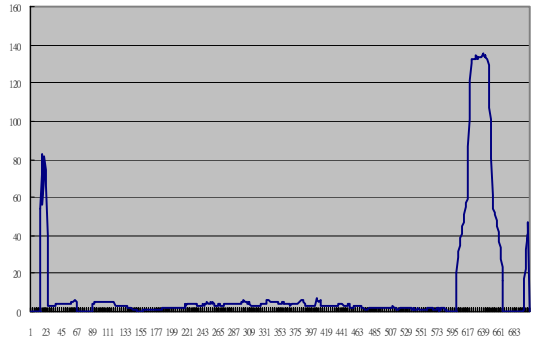

Figure 7. The result of the vertical projection of the image in Figure 6

The following are given heuristic rules for detecting the interesting events. The effectiveness of these heuristic rules will be reported based on the experimental results in Section 4.

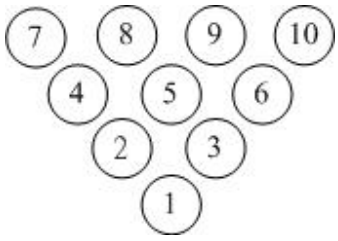

Figure 8. The location of the ten pins in the tenpin bowling

- *Bicycle*: It means there is a pin hidden behind another pin. It is difficult to detect bicycles even by human eyes. In our approach, it will be determined as a single pin first. However, the width of the bicycle must be larger than that of a true

[3] The area upon which the pins are set, also named plate.

single pin if their heights are the same. Therefore, the bicycle is detected if the ratio of width and height of a single pin is higher than that of a true single pin.

- *Bedposts*: It means the 7-10 leave. In the tenpin bowling, the distance between pins 7 and 10 is the largest among others. We detect bedposts using this characteristic.
- *Big ears*: It means the 4-6-7-10 leave. In the tenpin bowling, the 5-pin is in the middle of pins 4 and 6 and the two right or left pins are close. We detect big ears using this characteristic.

4. EXPERIMENT RESULTS

To show the effectiveness of the proposed methods in determining the result and the flooring and pinning frames, three experiments are performed. The data used in the experiments are taken from the professional bowling games on ESPN. The format of the videos is transformed to MPEG-2 format first.

There are 20 ball clips taken as the training data for setting up the thresholds, and then 509 ball clips are used as the testing data to evaluate the effectiveness. The results are classified into six classes. Table 1 shows the experimental results in a confusion matrix with the recall and precision, which are defined as:

$$\text{Recall} = \frac{\#\,\text{Correctly Retrieved}}{\#\,\text{Actual Total}} \times 100\,\%$$

$$\text{Precision} = \frac{\#\,\text{Correctly Retrieved}}{\#\,\text{Predicted Total}} \times 100\,\%$$

Table 1 The Confusion Matrix with Recall and Precision

Actual / Prediction	Strike	Bicycle	Bedposts	Big ears	Single Pin	Others	Predicted Total	Precision
Strike	239	0	2	0	10	19	270	88.52
Bicycle	1	5	0	0	16	2	24	20.83
Bedposts	6	0	10	0	3	5	24	41.67
Big ears	0	0	0	4	0	0	4	100
Single Pin	3	1	1	0	117	1	123	95.12
Others	12	0	0	0	11	41	64	64.06
Actual Total	261	6	13	4	157	68	509	
Recall	91.57	83.33	76.92	100	74.52	60.29		

Since the data are taken from professional games, the numbers of the strike or spare and the single pin are larger than those of the other classes. An excellent result appears in the big ears class, because these kinds of frames are clear and the threshold we chose is suitable for distinguishing big ears from others. Some confusion happens between the bicycle and single pin. This is because we determine the bicycle as the single pin first and use their different ratios of the width and the height of the pin to separate them out. The varying angles of the camera cause the threshold of the ratios hard to decide. If we try to get a higher recall in the bicycle class, the threshold will be unfavorable to the precision. Another major factor which causes the inaccuracy is the difference of alleys. The right and the left boundaries of each ROI are not considered since the effect of the zoom in and zoom out is not the same in all ball clips. Therefore, the noises in the right and left portions of the ROI are often falsely detected as the pins.

Table 2 The filter rate of strike or spare

	Total Num	FormerDetection	LaterDetection	Filter Rate
Strike or Spare	239	212	27	89%

In our approach, there are two steps for detecting if a result is a strike or a spare. If the result is detected in the former step, the later process can be ignored and the processing time saved. Table 2 shows the filter capability of our approach achieves at around 89 percents.

In Table 2, the **TotalNum** is the total number of the clips correctly detected as strike or spare, the **FormerDetection** is the number of the clips detected in the former step, the **LaterDetection** is the number of the clips detected in the later step, and the **FilterRate** is defined as

$$FilterRate = \frac{FormerDetection}{TotalNum} \times 100\%$$

Table 3 The Effectiveness of the determination of the two hits

	AcceptableNum	UnacceptableNum	AcceptableRate
Flooring Frame	141	98	58%
Pinning Frame	127	112	53%

Table 3 shows the results of the acceptable flooring frames and the pinning frames, where the acceptable rate is defined as:

$$\text{AcceptableRate} = \frac{\#\,\text{AcceptableNum}}{\#\,\text{AcceptableNum} + \#\,\text{UnacceptableNum}} \times 100\%$$

Whether the captured flooring frame or pinning frame is accurate is decided by users. When there are noises in the game and the ranges of the frequencies of the noises and the two hits overlap, it is difficult to capture the hits from the audio data. Therefore, both the acceptable rates are only between 50% and 60%.

5. CONCLUSION

A bowling information system is designed and implemented, which contains the video content information, game-related information and player information. All information is described by MPEG-7 Description Schemes. In addition, a semi-automatic annotator is designed to integrate both the manual conceptual feature annotation and the automatic perceptual feature extraction. Through a query interface, users retrieve any information they want to know about the bowling games. Our succeeding work is to study other perceptual features such as the scoreboard display and the slow motion in the video to more precisely derive the interesting events. The method for indexing XML files and efficient query processing in path expression will also be investigated. Furthermore, the framework will be applied to establish other kinds of sports information systems. The main difference between each sports information system is the

definition and extraction of the interesting events and the important frames.

6. ACKNOWLEDGMENTS

This work was partially supported by the MOE Program for Promoting Academic Excellent of Universities under the grant number 89-E-FA04-1-4.

7. REFERENCES

[1] A. Alatan, A. Akansu, and W.Wolf, "Multi-modal Dialog Scene Detection Using Hidden Markov Models for Content-based Multimedia Indexing," *Multimedia Tools and Applications,* Vol. 14, No. 2, pp.137-151, June 2001.

[2] A. Del Bimbo, Visual Information Retrieval. Morgan Kaufmann Publishers, 1999.

[3] D. Tjondronegoro and Y. P. Chen, "Content-Based Indexing and Retrieval Using MPEG-7 and X-Query in Video Data Management Systems," *World Wide Web Journal, Kluwer Academic Publishers*, Vol. 5, No. 3, pp. 207-227, 2001.

[4] M. R. Naphade and T. S. Huang, "A Probabilistic Framework for Semantic Video Indexing, Filtering, and Retrieval," *IEEE Transactions on Multimedia*, Vol. 3, No. 1, pp. 141-151, March 2001.

[5] S. Nepal, U. Srinivasan, and G. Reynolds, "Automatic Detection of 'Goal' Segments in Basketball Videos," *Proc. ACM Multimedia* 2001, pp. 261-269.

[6] S. Preiffer, R. Lienhart, and W. effelsbsrg, "Scene Determination Based on Video and Audio Feature," *Multimedia Tools and Applications,* Vol. 15, No. 1, pp. 59-81, September 2001.

[7] G. Sudhir, J. C. M. Lee, and 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*, 1998.

[8] V. Tovinkere and R. J. Qian, "Detecting Semantic Events in Soccer Games: Towards A Complete Solution," *Proc. IEEE International Conference on Multimedia & Expo* 2001.

[9] E. Wold, T. Blum, D. Keislar, and J. Wheaton, "Content-based Classification, Search, and Retrieval of Audio," *IEEE Multimedia,* Vol. 3, No. 2, pp. 27-36, 1996.

[10] ISO/IEC JTC1/SC29/WG11 N4509, "Overview of the MPEG-7 Standard", Pattaya, December 2001.

[11] ISO/IEC JTC 1/SC 29/WG 11/N3966, "Text of 15938-5 FCD Information Technology – Multimedia Content Description Interface – Part 5 Multimedia Description Schemes," Singapore, March 2001