Hindawi Publishing Corporation
EURASIP Journal on Applied Signal Processing
Volume 2006, Article ID 32135, Pages 1–13
DOI 10.1155/ASP/2006/32135

Multimodal Semantic Analysis and Annotation for Basketball Video

Song Liu, Min Xu, Haoran Yi, Liang-Tien Chia, and Deepu Rajan

School of Computer Engineering, Nanyang Technological University, Block N4, 02A-32, Nanyang Avenue, Singapore 639798

Received 1 September 2004; Revised 17 February 2005; Accepted 14 March 2005

This paper presents a new multiple-modality method for extracting semantic information from basketball video. The visual, motion, and audio information are extracted from video to first generate some low-level video segmentation and classification. Domain knowledge is further exploited for detecting interesting events in the basketball video. For video, both visual and motion prediction information are utilized for shot and scene boundary detection algorithm; this will be followed by scene classification. For audio, audio keysounds are sets of specific audio sounds related to semantic events and a classification method based on hidden Markov model (HMM) is used for audio keysound identification. Subsequently, by analyzing the multimodal information, the positions of potential semantic events, such as "foul" and "shot at the basket," are located with additional domain knowledge. Finally, a video annotation is generated according to MPEG-7 multimedia description schemes (MDSs). Experimental results demonstrate the effectiveness of the proposed method.

1. INTRODUCTION

In recent years, with the remarkable increase of video data generated and distributed through networks, there is an evident need to develop an intelligent video browsing and indexing system. To build such a system and facilitate content-based video accessing, automatic semantic extraction is a prerequisite and a challenge to multimedia-understanding systems. Therefore, semantic video analysis and annotation have received much interest and attracted research efforts. The previous research works [1–3] attempt to extract the semantics from visual and motion information. However, the investigation on extracting the semantic information from multimodal data is still very limited. In this paper, we develop tools based on visual, motion, and audio information for analyzing and annotating basketball video using both low-level features and domain knowledge. In particular, we show that the multimodal-based approach can generate reliable annotation for basketball video which cannot be successfully achieved using a single mode. We address the problem of semantic basketball video analysis and annotation for MPEG compressed videos using multimodal information. The problem has three related aspects: (1) analyze the structure of the basketball video, (2) locate the potential positions where an interesting event occurs, and (3) represent the results in an annotation file utilizing standardized descriptions. Since the semantic understanding of video content is highly dependent on the utilization of contextual information and domain rules, a basketball video analysis and annotation method is proposed based on visual, motion, and audio information as well as domain-specific knowledge.

Generally, the processing of sports video includes the following areas: analysis of the structure of video, detection of important events or activities, following a specific player's actions, and generating the summary. Video analysis aims to extract such semantic information within a video automatically. With such semantics, represented in terms of high-level descriptors, indexing, searching, and retrieving the video content can be improved. From the point of view of video processing using visual and motion information, several sports video analysis and modeling methods have been investigated. In [4], low-level soccer video processing algorithms and high-level event and object detection algorithms are utilized for automatic, real-time soccer video analysis and summarization. In [5], color and motion features are used for dominant scene clustering and event detection. However, the above methods do not take the motion information which is an important cue for sports video analysis into full consideration. In [6], the authors utilize the motion information for describing individual video object, but object segmentation for complex scenes like sports video is still a challenging problem. Thus, we propose an approach to differentiate camera motion and object motion from the total motion without object segmentation. In the proposed method, a modified scene detection algorithm based on both visual and motion prediction information is introduced. New motion features are proposed to capture the total motion, camera motion, and object motion, respectively. The camera motion is estimated from the motion vectors in the compressed video using an iterative algorithm with robust outlier rejection. The reasons for using motion features are twofold: (1) motion information has strong relationship with semantic event, that is, different events exhibit different motion patterns; (2) different events can be identified by motion features within a game and the video model generated by analyzing motion features is flexible enough to be applied in other classes of sports videos.

At the same time, audio information, which is an important type of media and also a significant part of video, has been realized as an important cue for semantics extraction. Most of the existing works try to employ audio-visual compensation to solve some problems which cannot be successfully solved only by visual analysis [7–10]. Nepal et al. [9] employed heuristic rules to combine crowd cheer, score display, and change in motion direction for detecting "Goal" segments in basketball videos. Han et al. [8] used a maximum entropy method to integrate image, audio, and speech cues to detect and classify highlights from baseball video. An event detection scheme based on the integration of visual and auditory modalities was proposed in [7, 10]. Recently, several frameworks [11, 12] for video indexing which support the multimodal features have been reported. However, they do not provide detailed descriptions about the implementation of multimodal system. To improve the reliability and efficiency in video content analysis, visual and auditory integration methods have been widely researched. Audio content analysis is the necessary step for visual and auditory integration. Effective audio analysis techniques can provide convincing results. In consideration of computational efficiency, some research efforts have been done for pure audio content analysis [13, 14]. Rui et al. [13] presented baseball highlight extraction methods based on excited audio segments detection. Game-specific audio sounds, such as whistling, excited audience sounds, and commentator speech, were used to detect soccer events in [14]. In this paper, we propose a new classification method based on hidden Markov model (HMM) to substitute our earlier methods [7, 10, 14] in which we used hierarchical support vector machine (SVM) to identify audio keysounds. The audio signals were segmented into 20-millisecond frames for frame-based identification while the audio signals are continuous time series signals rich in context information. By using SVM, we did not take into account the contextual information which is significant for time series classification. HMM is a statistical model of sequential data that has been successfully used in many applications including artificial intelligence, pattern recognition, speech recognition, and modeling of biological sequences [15]. Recently, HMM were introduced to sports video analysis domain [16–19]. Assfalg et al. [16] used HMM to model different events, where states were used to represent different camera motion patterns. In [18], Xie et al. tried to model the stochastic structures of play and break in soccer game with a set of HMMs in a hierarchical way. Dynamic programming techniques were used to obtain the maximum likelihood play/break segmentation of the soccer video sequence at the symbol level. These works demonstrated that HMM is an effective and efficient tool to represent continuous-time signals and discover structures in video content. However, to achieve detailed semantic basketball video analysis and annotation, we have combined the audio and motion features with other low-level features like color and texture.

Before ending this introduction, we list our main contributions: (1) motion-based scene boundary detection, (2) basketball scene classification based on visual and motion information, (3) HMM-based audio keysound detection, (4) high-level semantic inference and multimodal event detection, and (5) MPEG-7 standard compliant output for basketball video annotation. The paper is organized as follows. Section 2 describes video and audio processing for basketball video analysis and annotation, respectively. Section 3 presents the experimental results that quantify the performance of the proposed approach. Finally, conclusions are drawn in Section 4.

2. MULTIMODAL BASKETBALL VIDEO ANALYSIS AND ANNOTATION

The proposed multimodal video analysis consists of four components: (A) video segmentation and classification using visual and motion features, (B) audio keysound extraction based on HMM, (C) high-level semantic extraction and event detection utilizing multimodal information, and (D) annotation file generation. We explain the above components in the following subsections.

2.1. Video analysis utilizing visual and motion features

The proposed video analysis algorithm utilizing visual and motion features includes two stages: (1) shot and scene boundary detection and (2) scene classification. We will discuss these two stages in this subsection.

2.1.1. Shot and scene boundary detections

The shot and scene boundary detection is the initial step in our video analysis algorithm. Shot is the physical boundary of video, while scene is the semantic boundary of it [20, 21]. Although there is a rich literature of algorithms for detecting video shot and scene boundaries, it is still a challenging problem for basketball video. As mentioned above, scene can be viewed as a semantic unit. Unlike other types of videos, for example, movie, in which a scene is a group of shots which constitute the semantic unit, the scene in basketball video might be a segment of a shot. In basketball video, a single video shot could be a court-view camera that tracks the players or basketball for a significant amount of time without cuts or transitions but with plenty of panning and some zooming. Generally, one or many meaningful semantics, like actions or events, for example, *shot at the basket* or *foul*, are contained in such kinds of shots. Since it is hard to extract the detailed information for these actions or events from a single long camera shot, it is necessary to further partition the shot into scenes. Therefore, in current implementation, we segment a basketball video into shots first, and then further segment a shot into several scenes. After analyzing the structure of the long camera shot, we found that the semantics in the shot have strong relationship with the global motion associated with the movement of the camera. For example, actions such as *shot* and *foul* will occur most likely when the camera motion is slow, while the fast camera motion often indicates offensive and defensive exchange. Thus, we propose a video temporal segmentation algorithm based on color and motion prediction information to detect shot and scene boundaries simultaneously.

Use of motion prediction information in MPEG video to detect the shot boundary has been proposed in [22]. Motion vectors are divided into four types and the number of each type of macroblocks (MBs) in a frame is used to indicate the similarity/dissimilarity of that frame with its neighboring frames. In our current algorithm, we extend the method in [22] to combine a color-based shot boundary detection method to detect the shot/scene boundaries in basketball videos simultaneously.

In the first step, we use the difference between the color histograms of the neighboring frames, D_h, as the feature to detect shot boundaries, which is defined as

$$D_h = \frac{\sum_{i=0}^{N} \left| H_n(i) - H_{n-1}(i) \right|}{\text{width} \cdot \text{height}} > T_g, \tag{1}$$

where N is the number of bins in the color histogram, H_n and H_{n-1} are the color histograms of frames n and $n-1$, respectively, width · height denotes the pixel numbers in each frame, and T_g is the threshold for detecting an isolated sharp peak in a series of discontinuity values of D_h.

Subsequently, to detect the scene boundary accurately, we modified the original algorithm defined in [22]. Firstly, we modified the definition of frame dissimilarity ratio (FDR) to provide a precise scene boundary detection. The new FDR is defined as

$$\text{FDR}_n = \begin{cases} \dfrac{Fw_{n-1}}{Bi_{n-1}} & \text{for I-frame,} \\ \dfrac{In_n}{Fw_n} & \text{for P-frame,} \\ \dfrac{\left| Fw_n - Bk_n \right|}{Bi_n} & \text{for B-frame,} \end{cases} \tag{2}$$

where In, Fw, Bk, and Bi represent the number of the MBs for intracoded, forward predicted, backward predicted, and bidirectionally predicted frames, respectively, and n denotes the frame number. We modified the FDR by (1) creating an expression of FDR for P-frame to provide more accurate feature description when the boundary is located at P-frames, and (2) modifying the expression of FDR for B-frame to eliminate false detection if $Fw_n \approx Bk_n$ and they are all much larger than Bi_n. Consider the following frame structure in an MPEG bit stream: …I_1 B_2 B_3 B_4 P_5 B_6 B_7 B_8 P_9…. If the shot change takes place at B_3, FDRs for B_2, B_3, and B_4 will be very high. In order to determine the exact location of the shot boundary, we observe that B_2 is mostly forward predicted while B_3 and B_4 are mostly backward predicted. Thus, at the shot boundary there is a change in the dominant MB type of the B-frame. So, we define a parameter called dominant MB change (DMBC_n) for frame n as

$$\text{DMBC}_n = \begin{cases} 0 & \text{if } \left(Bk_{n-1} - Fw_{n-1}\right) > 0, \\ & \left(Bk_{n+1} - Fw_{n+1}\right) \le 0 \qquad \text{for I-frame,} \\ 1 & \text{otherwise} \qquad \text{for I-frame,} \\ 1 & \qquad \text{for P-frame,} \\ 0 & \text{if } \left(Bk_n - Fw_n\right) \\ & * \left(Bk_{n-1} - Fw_{n-1}\right) > 0 \qquad \text{for B-frame,} \\ 1 & \text{if } \left(Bk_n - Fw_n\right) \\ & * \left(Bk_{n-1} - Fw_{n-1}\right) \le 0 \qquad \text{for B-frame.} \end{cases} \tag{3}$$

Thus, the DMBC acts a filter to locate the scene boundary precisely.

2.1.2. Scene classification

We classify basketball scenes into six classes: (1) fast-motion court-view scenes, (2) slow-motion court-view scenes, (3) penalty scenes, (4) in-court medium scenes, (5) out-of-court or close-up scenes, and (6) bird-view scenes. The definitions and characteristics of each class are given below.

(i) *Fast-motion court-view scene*. This scene displays a global view of the court and has obvious global motion; hence, this type of scene can serve to differentiate the offensive and defensive exchange between the teams.

(ii) *Slow-motion court-view scene*. A scene that displays the global view of the court and has insignificant global motion; hence, this type of scene can be used to locate the interesting events.

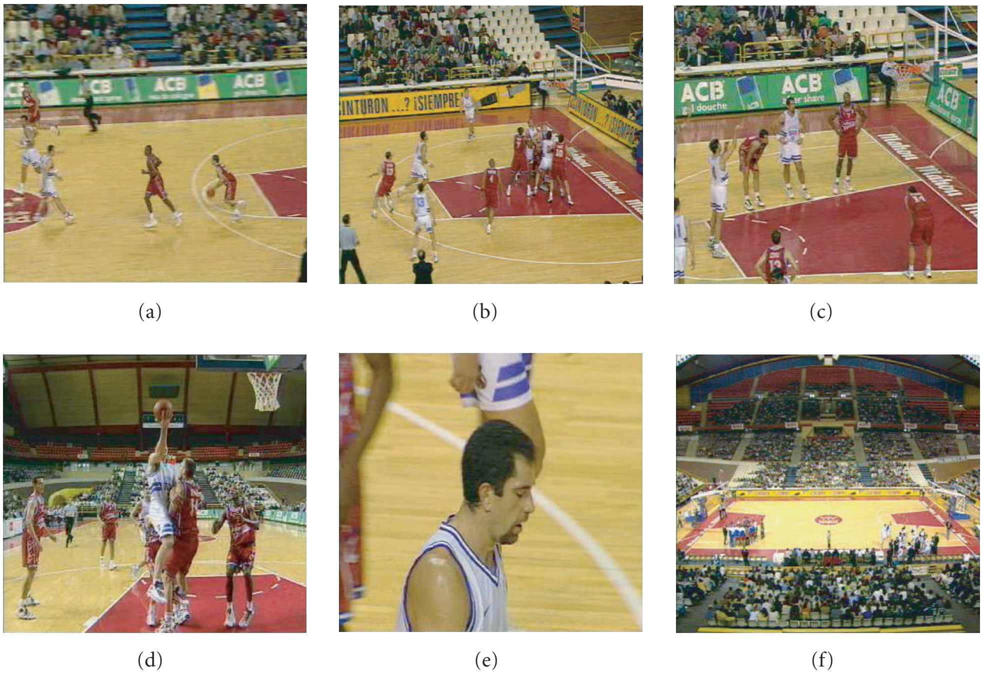
(iii) *Penalty scene*. A scene that shows the taking of a penalty under the rim.

(iv) *In-court medium scene*. A scene that focuses on a whole player or players in a cluster. Generally, it is a zoomed-in court-view scene. In most cases, a replay is shown as in-court medium scene.

(vi) *Out-of-court or close-up scene*. Such scenes display the audience, coach, and close-ups. These types of scenes usually indicate a break in the match or highlight the player who has just executed an exciting event.

(vii) *Bird-view scene*. A scene that shows a global view of the whole gymnasium and is usually taken from a stationary camera.

Figure 1 shows an example each of the six typical scenes. A series of texture and motion features are extracted for classifying a scene into one of the above six classes. In our initial experiment, the texture features were extracted from the key frame of a scene, which is an I-frame located at the centre of the scene. Two texture features, *run-length feature* [23] and



(a) (b) (c)

(d) (e) (f)

Figure 1: Example of typical scenes. (a) Fast-motion court-view scene. (b) Slow-motion court-view scene. (c) Penalty scene. (d) In-court medium scene. (e) Out-of-court or close-up scene. (f) Bird-view scene.

co-occurrence feature [24], are generated from the key frame. The run-length feature vector has four dimensions, namely, long-run emphasis, shot-run emphasis, run-length entropy, and run-length energy. The co-occurrence feature vector has three dimensions—contrast, energy, and entropy.

The second kind of features are motion features. In order to differentiate the camera motion and object motion from the total motion, we need to estimate the global motion. Model-based motion estimation has been reported extensively in literature [25]. In [26], the affine parameter estimation problem is formulated as a nonlinear minimization problem which is solved using an iterative algorithm. The objective function to be minimized is the sum of square difference between the original image and the warped image by the affine transform parameters. This method is semiautomatic because the user needs to identify at least three corresponding feature points in two frames.

Our global motion estimation algorithm is an iterative algorithm with robust outlier rejection. The affine parameters are chosen so as to fit the block-based motion vector between two frames which are available from the MPEG compressed video stream. We model the global motion as

$$\begin{aligned} mvx_i &= p_1 x_i + p_2 y_i + p_3, \\ mvy_i &= p_4 x_i + p_5 y_i + p_6, \end{aligned} \tag{4}$$

where mvx_i and mvy_i are the components of the motion vector for a particular macroblock (MB), x_i and y_i are the coordinates of the center of the MB, and p_i's are the affine parameters that we call *motion vector affine parameters*.

We define a coordinate row vector $\mathbf{c_i}$ for block i as $\mathbf{c_i} = (x_i, y_i, 1)$. Next, the coordinate matrix C is formed by vertically concatenating the row vectors $\mathbf{c_i}$ for all blocks which are *not marked as outliers*. C is, then, an $N \times 3$ matrix, where N is the number of macroblocks not marked as outliers. The vectors $\mathbf{V_x}$ and $\mathbf{V_y}$ are formed by collecting all the mvx_i and mvy_i, respectively, for the MBs not marked as outliers. Lastly, the motion vector affine parameters are grouped together as $\mathbf{p_x} = (p_1, p_2, p_3)^T$ and $\mathbf{p_y} = (p_4, p_5, p_6)^T$. From these definitions, we can write $\mathbf{V_x} = C\mathbf{p_x}$ and $\mathbf{V_y} = C\mathbf{p_y}$ which are then solved for $\mathbf{p_x}$ and $\mathbf{p_y}$ using the pseudoinverse matrix of C:

$$\begin{aligned} \mathbf{p_x} &= \left(C^T C\right)^{-1} C^T \mathbf{V_x}, \\ \mathbf{p_y} &= \left(C^T C\right)^{-1} C^T \mathbf{V_y}. \end{aligned} \tag{5}$$

After each iteration, we calculate the residual motion vector R_{mv_i} as the absolute difference between the actual motion vector and the estimated motion vector, that is, $R_{mv_i} = |(mvx_i - mvx_i') - (mvy_i - mvy_i')|$, where mvx_i' and mvy_i' are the estimated components of the motion vector for macroblock i. We propose an adaptive threshold mechanism to reject outliers in the residual motion vectors. The threshold T is decided by comparing the mean of the residual motion vectors over all MBs with a small constant α and choosing the maximum of the two, that is, $T = \max(\text{mean}\,(R_{mv_i}), \alpha)$. The role of α is to prevent the rejection of a large number of motion vectors if the mean of the residuals is very small. We choose α to be equal to 0.5. The algorithm is initialized by labeling all macroblocks as *inliers*.

Having determined the frame-by-frame global motion, we now describe our motion features for shot classification.

The first class of motion features is *global camera motion description* that includes *camera horizontal motion* (CHD), *camera vertical motion* (CVD), and *camera zoom* (CZD). The above features are defined as

$$\begin{aligned} \text{CHD} &= \frac{\sum_{i=0}^{N} p_{3i}}{N}, \\ \text{CVD} &= \frac{\sum_{i=0}^{N} p_{6i}}{N}, \\ \text{CZD} &= \sum_{i=0}^{N} \frac{p_{1i} + p_{5i}}{2}, \end{aligned} \tag{6}$$

where N is the number of frames included in a scene.

The second class of motion features are *total motion matrix*, *object motion matrix*, and *camera motion matrix*, which describe the amounts of total motion, object motion, and camera motion for each macroblock. These features are defined as

$$\begin{aligned} tmx_i &= |mvx_i|, \qquad tmy_i = |mvy_i|, \\ cmx_i &= \min(|mvx_i|, |gmvx_i|), \\ cmy_i &= \min(|mvy_i|, |gmvy_i|), \\ omx_i &= \max(0, |mvx_i| - |gmvx_i|), \\ omy_i &= \max(0, |mvy_i| - |gmvy_i|), \end{aligned} \tag{7}$$

where gmv (denoting $gmvx_i$, $gmvy_i$) is the global motion vector filed that is constructed at each macroblock with its centroid coordinates. Since the estimated "$|gmv|$" may be larger than "$|mv|$" (denoting mvx_i, mvy_i), we use cm (denoting cmx_i, cmy_i) as a minimum of $|mv|$ and $|gmv|$. Similarly, if $|gmv|$ is larger than $|mv|$, the om (denoting omx_i, omy_i) may be negative since $om = tm - cm$. However, the amount of motion should not be negative. Therefore, we choose the maximum of either 0 or the difference. Then we accumulate tm, om, and cm across a shot. Total motion (TM), camera motion (CM), and object motion (OM) for a scene, k, with n number of frames, and r MVs for each frame, are defined as

$$\begin{aligned} TMX_k &= \sum_{l=1}^{n}\sum_{i=1}^{r} tmx_{li}, \\ TMY_k &= \sum_{l=1}^{n}\sum_{i=1}^{r} tmy_{li}, \\ CMX_k &= \sum_{l=1}^{n}\sum_{i=1}^{r} cmx_{li}, \\ CMY_k &= \sum_{l=1}^{n}\sum_{i=1}^{r} cmy_{li}, \\ OMX_k &= \sum_{l=1}^{n}\sum_{i=1}^{r} omx_{li}, \\ OMY_k &= \sum_{l=1}^{n}\sum_{i=1}^{r} omy_{li}, \end{aligned} \tag{8}$$

where TM (denoting TMX_k, TMY_k), CM (denoting CMX_k, CMY_k), and OM (denoting OMX_k, OMY_k) can be represented as a matrix of r products. Finally, the projection values of TM, CM, and OM on the horizontal and vertical directions are used as feature vectors in our experiment.

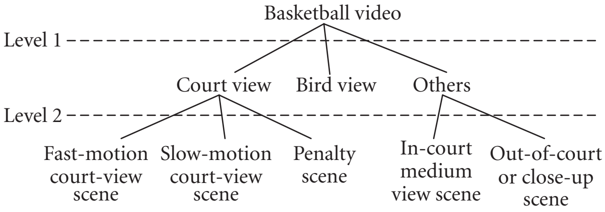

Basketball video
Level 1
Court view
Bird view
Others
Level 2
Fast-motion court-view scene
Slow-motion court-view scene
Penalty scene
In-court medium view scene
Out-of-court or close-up scene

Figure 2: The structure of hierarchical SVM.

A hierarchical SVM classifier is built for basketball scene classification. The structure of the classifier is shown in Figure 2. The basketball video scenes were classified into three classes, *court view*, *bird view*, and *others* utilizing the texture-based features at the first run of the SVM classifier. The class *courts view* was further divided into three classes, *fast-motion court-view scene*, *slow-motion court-view scene*, and *penalty scene*, based on the global camera motion information and TM, CM, OM matrix. Lastly, the class *others* was divided into two classes, *in-court medium scene* and *out-of-court or close-up scene* based on the texture information and TM, CM, OM matrix.

2.2. Audio keysound detection utilizing hidden Markov models

Audio keysounds are defined as some specific audio sounds which have strong hints to interesting events. Especially in sports video, some game-specific audio sounds (e.g., whistling, excited commentator speech, etc.) have strong relationships with the actions of players, referees, commentators, and audience. These audio sounds may take place in the presence of interesting events as listed in Table 1. Generally, excited commentator speech and excited audience sounds play important roles in highlight detection of sports video. Other keysounds may be specific to a kind of sports game. Audio signal exhibits the consecutive changes in values over a period of time, where variables may be predicted from earlier values. That is, strong context exists. In consideration of the success of HMM in speech recognition, we propose our HMM-based audio keysounds detection system. The proposed system includes three stages, which are feature extraction, data preparation, and HMM learning, as shown in Figure 3. As illustrated in Figure 3, selected low-level features are firstly extracted from audio streams and tokens are added to create observation vectors. These data are then separated into two sets for training and testing. After that, HMM is trained then reestimated by using dynamic programming. Finally, according to maximum posterior probability, the audio keysound with the largest probability is selected to label the corresponding testing data. We next introduce the details of the proposed system in the following.

2.2.1. Feature extraction

We segment audio signal at 20 milliseconds per frame which is the basic unit for feature extraction. Mel-frequency

Table 1: Audio keysounds' relationship to potential events.

Sports	Audio keysounds	Potential events
Tennis	Applause	Score
	Commentator speech	At the end (or the beginning) of a point
	Silence	Within a point
	Hitting ball	Serve, ace, or return
Soccer	Long whistling	Start of free kick, penalty kick, or corner kick, game start or end, offside
	Double whistling	Foul
	Multiwhistling	Referee reminding
	Excited commentator speech or excited audience sound	Goal or shot
	Plain commentator speech or plain audience sound	Normal
Basketball	Whistling	Foul
	Ball hitting backboard or basket	Shot
	Excited commentator speech or excited audience sounds	Fast break, drive, or score
	Plain commentator speech or plain audience sound	Normal

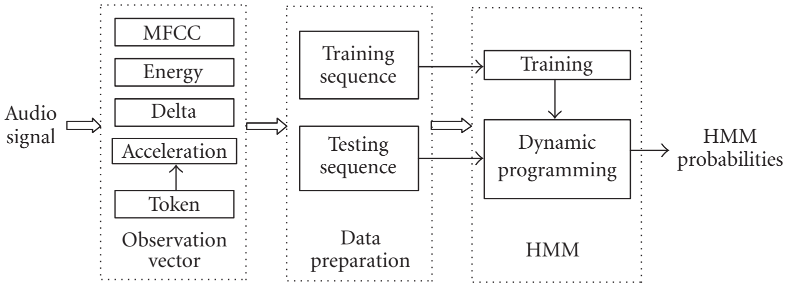


Figure 3: Proposed audio keysounds detection system.

cepstral coefficient (MFCC) and energy are selected as the low-level audio features as they are successfully used in speech recognition and further proved to be efficient for audio keysound detection in [14]. Delta and acceleration are further used to accentuate signal temporal characters for HMM [27].

Mel-frequency cepstral coefficient

The mel-frequency cepstrum is highly effective in audio recognition and in modeling the subjective pitch and frequency content of audio signals. Mel scale is calculated as

$$\text{Mel}(f) = 2595 \times \log_{10}\left(1 + \frac{f}{700}\right), \tag{9}$$

where $\text{Mel}(f)$ is the logarithmic scale of the normal frequency scale f. Mel scale has a constant mel-frequency interval, and covers the frequency range of 0–20050 Hz. The mel-frequency cepstral coefficients (MFCCs) are computed from the FFT power coefficients which are filtered by a triangular bandpass filter bank. The filter bank consists of 12 triangular filters. The MFCCs are calculated as

$$C_n = \sqrt{\frac{2}{k}} \sum_{k=1}^{K} (\log S_k) \cos\left[n(k-0.5)\frac{\pi}{k}\right], \quad n = 1, 2, \ldots, N, \tag{10}$$

where $S_k (k = 1, 2, \ldots, K)$ is the output of the filter banks and N is total number of samples in a 20-millisecond audio unit.

Energy

The energy measures amplitude variations of the speech signal. The energy is computed as the log of the signal energy, that is, for audio samples s_n, where $\{n = 1, 2, \ldots, N\}$:

$$E = \log \sum_{n=1}^{N} {s_n}^2. \tag{11}$$

Delta and acceleration

Delta and acceleration effectively increase the state definition by including first- and second-order memory of past states. The delta and acceleration coefficients are computed using the following simple formula (t means the tth coefficient in feature vector):

$$\delta(C_t) = C_t - C_{t-1}; \qquad \text{ACC}(C_t) = \delta(C_t) - \delta(C_{t-1}). \tag{12}$$

2.2.2. *Our proposed hidden Markov model*

As for the HMM generation, we need to determine the HMM topology and statistical parameters. In this research,

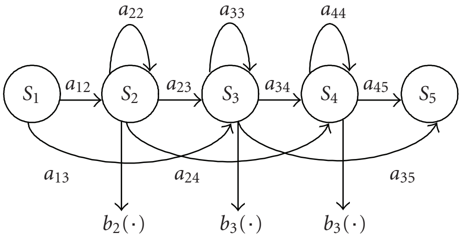


FIGURE 4: The left-right HMM with 5 states.

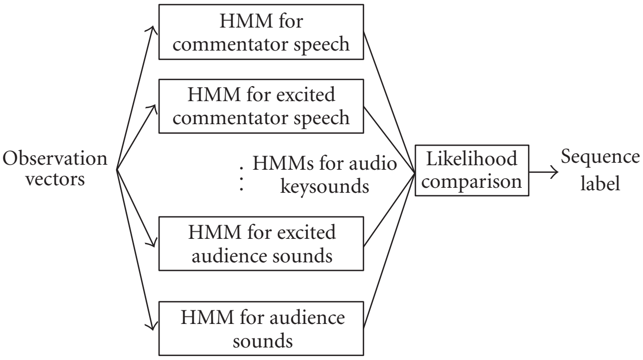


FIGURE 5: The HMM overview structure.

we choose the typical left-right HMM structure, as shown in Figure 4, where $S = \{s_1, \dots, s_5\}$ are five states; $A = \{a_{ij}\}$ are the state transition probabilities; and $B = \{b_i(v_k)\}$ are the observation probability density functions which are represented by a mixture Gaussian density. In our case, each audio frame is regarded as one observation. We use $\lambda = (\prod, A, B)$ to denote all the parameters, where $\prod = \{\pi_i\}$ are the initial state probabilities. In the training stage, observation vectors are separated into classes to estimate the initial value of B firstly. Then, to maximize the probability of generating an observed sequence, that is, to find $\lambda^* = \arg\max_\lambda p(O \mid \lambda)$, we use Baum-Welch algorithm to adjust the parameters of model λ. The recognition stage is shown in Figure 5, where each audio keysound is associated with a pretrained HMM. For each incoming audio sample sequence $A = \{f_1, f_2, \dots, f_l\}$ containing f audio frames, the resulting audio features from each frame form the observation vectors. Later, the likelihood of every HMM is computed. The audio sequence A is recognized as keysound k, if $P(O \mid \lambda_k) = \max_l P(O \mid \lambda_l)$ [27]. In the next step, we are concerned about two issues. First is the number of states that are suitable for an HMM. The other is the HMM sample length selection. We will discuss these two issues in Section 3.

2.3. Multimodal structure analysis and event detection

Utilizing the video and audio analysis algorithms described earlier, we have achieved high-level scene classification and audio keysound detection. The next step is to combine the visual information and audio keysounds to infer a higher level of semantic understanding, for example, detecting the positions of interesting events. These interesting events inside the basketball game include foul, steal ball, shot at the basket, and so forth. The goal of proposed event detection program is to locate the positions of the events and label a scene with an event.

As mentioned in Section 2.2, the audio keysound detection algorithm can detect several audio keysounds that indicate the potential events. However, we cannot locate all events precisely by only using audio information. For example, we cannot distinguish the whistling due to break or foul if only audio information is provided. At same time, although the types and orders of changes in the scenes generated by scene classification algorithm provide us with a good understanding of the structure of a basketball video, it is still very hard to detect the events using only visual features. In order to locate the exact scenes where events occur, we propose a multimodal event detection mechanism to get benefits from both visual and audio information. From the domain knowledge of the basketball game, we know that the locations of events have strong relationships with the camera movement and position. The global camera motion provides useful information for event detection, because the camera tracks the players or basketball during the game. Most of the events are located at the scenes with small camera motion. Also, the amount of camera motion in the next scene indicates what kind of events may occur in the current scene. To measure the amount of camera motion inside a scene precisely, we define a feature called modified accumulated camera motion in time (MACM) as the product of accumulated camera motion in time (ACM) and dominant camera motion filter (DCMF), that is,

$$\text{MACM} = \text{ACM} \times \text{DCMF}, \tag{13}$$

where,

$$\text{ACM} = \begin{cases} (\text{CHD} - \text{CVD}) \cdot e^{-\text{CZD}} \cdot D_s & \text{if CHD} \cdot \text{CVD} > 0, \\ (\text{CHD} + \text{CVD}) \cdot e^{-\text{CZD}} \cdot D_s & \text{if CHD} \cdot \text{CVD} < 0, \end{cases} \tag{14}$$

where the D_s is the time duration for a single scene, and DCMF is used to filter out the conflict when two neighboring scenes have the same camera motion direction within a single shot:

$$\text{DCMF} = \begin{cases} 1 & \text{The first large camera motion} \\ & \text{scene in the long court-view shot,} \\ 1 & \text{if } (\text{ACM}_{\text{previous}}) \\ & *(\text{ACM}_{\text{current}}) < 0, \\ 0 & \text{All out-of court} \\ & \text{view scenes and others,} \end{cases} \tag{15}$$

where $\text{ACM}_{\text{previous}}$ indicates the previous detected large camera motion scene. If MACM is above the T_α, this scene has large camera motion. Based on the above definitions, we can classify scenes into two groups. We name a scene as offensive

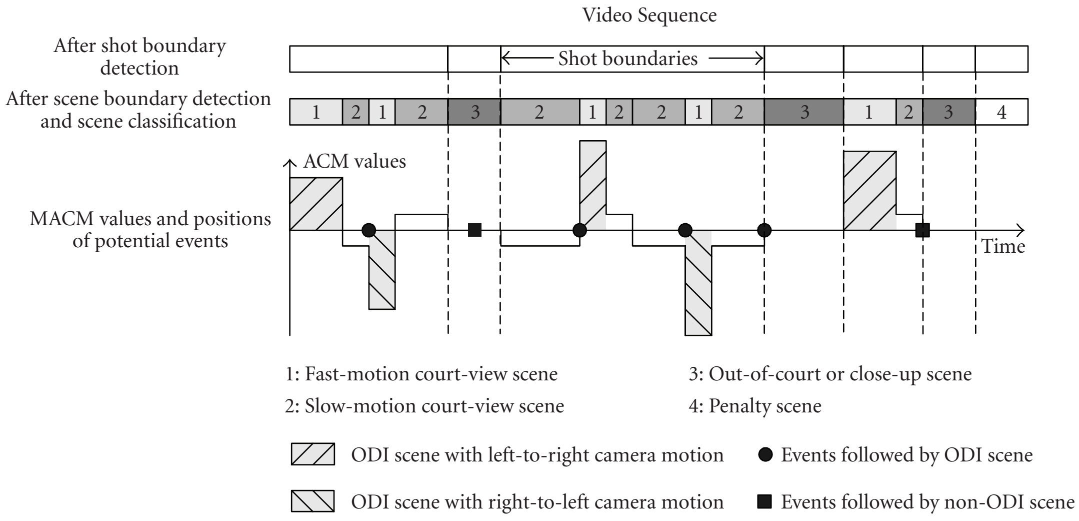

Video Sequence
After shot boundary detection
Shot boundaries
After scene boundary detection and scene classification
1 2 1 2 3 2 1 2 2 1 2 3 1 2 3 4
ACM values
MACM values and positions of potential events
Time
1: Fast-motion court-view scene
2: Slow-motion court-view scene
3: Out-of-court or close-up scene
4: Penalty scene
ODI scene with left-to-right camera motion
ODI scene with right-to-left camera motion
Events followed by ODI scene
Events followed by non-ODI scene

Figure 6: Example for ODI detection and event detection.

and defensive exchange interval (ODI) scene, if it contains large camera motion (MACM > T_α). Otherwise, we call it a non-ODI scene. Based on the sign of MACM value, two kinds of ODI scenes can be distinguished: ODI scene with left-to-right camera motion and ODI scene with right-to-left camera motion. In basketball video, some ODI scenes may not be captured by the camera because camera might focus on a single player when he/she is on the move. To detect these noncaptured ODI scenes, the detected ODI scene sequence is refined further. The refinement is based on the observation that left-to-right change and right-to-left change should alternate in the video. We assign a scene to be an ODI scene if there is a court-view scene between two scenes that have the same camera motion direction; otherwise we assign the second of the two scenes as non-ODI scene.

After ODI scene detection, we describe how to locate the positions of events using ODI information. Consider two categories of events: (1) *shot at the basket*, *steal ball*, and *offensive foul* and (2) *defensive foul*. For the events in category (1), they occur followed by an ODI scene and for the event in category (2), it occurs followed by a non-ODI scene. Therefore, we define these two categories of events as *events followed by ODI scene* and *events followed by non-ODI scene*. Since we have given how to detect ODI scene, locating the position for these two kinds of events can be achieved. Figure 6 shows an example of ODI detection. In the figure, the hashed blocks represent ODI scenes and the black squares or round dots represent the points where events occur. For example, event *foul* is an event occurring before the ODI scene, so we place a round dot on scene boundaries followed by a ODI scene. However, to present the time interval of one event, we use the scene before the boundary where event occurs to represent the event. After locating the potential positions of events using visual information, it is easy to combine the audio information to finally label a scene with an event. An algorithm for detection of "foul" and "shot at the basket" based on visual, audio information and some heuristic decision rules derived from the domain knowledge of basketball game is shown in Algorithm 1.

```
Input: Shot classification, ODI information and audio keysound
Output: The event label "foul" and "shot at the basket" for a scenes
if The current scene is "court-view and non-ODI scene" or (the current scene is ODI scene and it's neighbor scenes are non-court view scene) then
  if The audio keysound "whistling" has been detected and it does not occur at the beginning of the scene then
    if The next "court view scene" is an ODI scene then
      Event "offensive foul" detected
    else if The "whistling" occurs followed by non-ODI scene or penalty scene then
      Event "defensive foul" detected
    end if
  else if The audio keysound "excited sound" has been detected in this scene and the next "court view scene is an ODI scene. then
      Event "shot at the basket" detected
  end if
end if
```

Algorithm 1: Event detection.

2.4. MPEG-7 compliant annotation file generation

The objective of designing a video analysis and annotation system is to facilitate content-based search and retrieval of video entities. Thus, we need to store the results of processing and information in a highly structured format, which enables the annotation information to be queried and retrieved easily. MPEG-7 is a new multimedia standard, designed for describing multimedia content by providing a rich set of standardized descriptors and description schemas. The objective of the MPEG-7 standard is to allow interoperable searching, indexing, filtering, and access of multimedia content by enabling interoperability among devices that deal with multimedia content description [28]. The multimedia description schemes (MDSs) expand on the MPEG-7 descriptor by combining descriptors and other description schemes to provide description for both the immutable metadata and the content of audio, visual, and textual data. MPEG-7 MDSs consist of the following areas: basic elements, content description, content management, content organization, navigation, and access. We utilize the description schemes (DSs) of content management and description provided by MPEG-7 MDSs to represent the results of the proposed semantic basketball video analysis and annotation system. Currently, the results for video analysis, event detection, and audio keysounds detection are stored in two XML files based on different temporal decompositions of the video. The first file is utilized to store the video analysis and event detection information. There are two types of descriptors stored in this XML file, in which one type is manually annotated information and the other is automatically generated information. The manually annotated information includes the following.

(i) *CreationInformation DS*. That describes creation and production of the multimedia content. CreationInformation DS is composed of one Creation DS which contains information about the creation and production of the content not perceived in the content, such as author, director, and characters.

(ii) *TextAnnotation DS*: This DS contains a FreeTextAnnotation DS which provides a free text annotation for the video contents.

The automatically generated information includes the following.

(i) *MediaLocator DS*. It contains a MediaUri DS to describe the location of external media data.

(ii) *MediaTime DS*. This DS is utilized to specify the time intervals of a video segment. It contains MediaTimePoint DS and MediaDuration DS which describe a time point using Gregorian date and day time, and duration of a time period according to days and day time, respectively. By utilizing these two DSs, the location of one video segment in the whole video can be specified by time.

(iii) *AudioVisual DS*. This DS is utilized to describe the temporal decomposition of a video entity. To provide a highly structured description for the video contents, the AudioVisualSegment DS is used to describe segments of audio-visual content and their attributes and structural decompositions. The structure of the content description can be described as follows: one AudioVisual DS represents one audio-visual entity and contains one TemporalDecomposition DS, in which it contains several AudioVisualSegment DSs to represent the shots composing an audiovisual entity. Similarly, under each AudioVisualSegment DS describing the shot, there is a TemporalDecomposition DS also, and the AudioVisualSegment DS under this level of TemporalDecomposition DS describes the temporal decomposition of scenes.

In each TemporalDecomposition DS some attributes are generated automatically to describe a shot or scene; they are as follows.

(i) *MediaTime DS*. It specifies the time intervals of a video segment.

(ii) *Term DS*. It contains a termID tag to describe the scene classification information. The termID is defined in a ClassificationScheme DS based on the classification rules described in Section 2.1.2.

(iii) *Event DS*. It describes an event, which is a semantic activity that takes place at a particular time or in a particular location.

(iv) *FreeTextAnnotation DS*. It is utilized to describe ODI in a scene.

By using the DSs described above, all results of video analysis and event detection can be represented in a standardized and highly structured format. The second XML file is used to describe the information of audio keysounds detection. The difference between the two XML files is that they have different temporal decompositions. In the first XML file, the temporal decomposition is based on the results of shot and scene detection, and in the second XML file, the temporal decomposition is based on the intervals of audio keysounds. Since audio-visual segments in our current system have fixed time intervals, we calculate the time duration of a keysound by summing up the time for the neighboring segments labelled as having the same audio keysound before generating the annotation file. That audio keysound information is then represented using MPEG-7 DS in a tree structure similar to the XML file described earlier.

3. EXPERIMENTAL RESULT

In this section, we present the results of our algorithm for shot/scene detection, scene classification, audio keysound detection, and event detection. The test videos are two basketball videos from different matches with a total length of fifty minutes. The frame structure of the MPEG compressed test videos follows the standard GOP.

3.1. Video shot and scene detection

The performance of the algorithm for hard-cut boundaries and gradual transitions is tabulated together. In the test videos, wipes and dissolves were utilized in the replay and close-up shot. Overall, the algorithm achieves 84.3% recall and 97.5% precision rates over 286 shot boundaries. We got a low recall rate since our color-based shot detection algorithm could not detect the gradual transitions accurately. However, the scene detection algorithm helps to reduce the nondetection of gradual transitions. Since scene detection is a very important stage for generating the data which are utilized in the scene classification and semantic video analysis algorithms,

Table 2: Classification rates for level-1 and level-2 classes.

Classes	Correct classification rate (%)
Court view	95.2
Bird view	99.8
Others	95.0
Penalty scene	83.4
Slow-motion court-view scene	87.0
Fast-motion court-view scene	91.3
In-court medium view scene	88.0
Out-of-court or close-up scene	88.2

the results of these subsequent algorithms can be used to measure the performance of scene boundary detection indirectly.

3.2. Video scene classification

Currently, a two-class SVM classifier was implemented to handle the scene classification. For the case of multiple-class classification, the classification rate of target class versus others is used as the experimental results. Table 2 shows the results of scene classifications for the level-1 and level-2 scenes over a total of 1053 scenes. In the experiments, half of data set were used as training set and the remainder were used as test set.

3.3. Audio keysound detection

Excited commentator speech and excited audience sounds directly correspond to sports highlight which attracts audience's interests mostly. Compared with whistling and hitting the ball, the recognition of these two keysounds is quite challenging as excited parts always interlace with plain parts. Therefore, in our experiments, we concentrate on excited commentator speech and excited audience sounds.

The audio samples come from 40 minutes of basketball game. They are collected with 44.1 kHz sample rate, stereo channels, and 16 bits per sample. We used two third of the samples for training and one third for testing. For the HMM learning, different number of states may model different states transition process, which could influence the results. Moreover, as each kind of audio keysound has its own duration, we need to choose appropriate sample length for training different keysounds. Therefore, we conduct some experiments to compare HMM structures with various states and change HMM sample length to achieve the best performance of our proposed audio keysound detection system.

HMM with different hidden states

Table 3 shows the precision and recall rates for each audio keysound as the number of states are changed from 3 to 5. We find that 3-state HMM is good while 4-state HMM provides better performance for excited commentator. In some sports games, when the environment is very noisy, we cannot detect sports highlights only by excited audience sounds while excited commentator speech is able to provide the most important cues. Therefore, higher performance of excited commentator speech identification is necessary. Based on the above criteria and performance results, we thus use the 4-state HMM to generate audio keysounds.

Table 3: Performance of various HMMs with different states for audio keysound detection.

Audio keysounds	States number	Recall (%)	Precision (%)
Audience	5 states	95.74	95.74
	4 states	95.74	95.74
	3 states	100	100
Commentator	5 states	100	91.07
	4 states	98.04	94.34
	3 states	100	92.73
Excited audience	5 states	85.71	85.71
	4 states	85.71	85.71
	3 states	100	100
Excited commentator	5 states	66.67	100
	5 states	66.67	100
	4 states	86.67	100
	3 states	73.33	100

HMM with different sample lengths

Observation of real sports games reveals that the shortest keysound whistling lasts slightly longer than 0.2-second. Therefore, we segment audio signals into 0.2-second samples for whistling detection. However, other audio keysounds, such as commentator speech, excited audience sounds and so forth, last much longer than 0.2 second. Table 4 lists the results of different sample lengths for several types of audio keysounds. The results show that 1-second sample length is much better than 0.2 second for audience sounds and commentator-speech-related audio keysound detection. The main reason is that longer sample length provides much more contextual information for HMM to learn in order to differentiate between different audio keysounds.

Comparison between HMM and SVM

We perform a comparison between the HMM-based method and the SVM-based method [10]. According to the previous experimental results, 4-state left-right structure is selected to build HMM. We choose 0.2 second as sample length for whistling detection and 1 second for other audio keysounds (i.e., commentator speech, audience sounds, etc.). Compared with SVM-based audio keysound detection, the proposed HMM-based method achieves better performance as listed in Table 5. For the excited keysounds detection, which are more significant for highlight detection, the recalls and precisions are improved by at least 5%.

Table 4: Performance of different sample lengths for audio keysound detection (5-state HMM).

Audio keysounds	Sample length	Recall (%)	Precision (%)
Audience	0.2 s	95.39	96.61
	1 s	95.74	95.74
Commentator	0.2 s	96.52	83.33
	1 s	100	91.07
Excited audience	0.2 s	83.33	75.95
	1 s	85.71	85.71
Excited commentator	0.2 s	31.65	73.53
	1 s	66.67	100

Table 5: Audio keysound detection results (HMM versus SVM).

Audio keysounds	Methods	Recall (%)	Precision (%)
Whistling	SVM	99.45	99.45
	HMM	100	100
Audience	SVM	83.71	79.52
	HMM	95.74	95.74
Commentator	SVM	79.09	78.27
	HMM	98.04	94.34
Excited audience	SVM	80.14	81.17
	HMM	85.71	85.71
Excited commentator	SVM	78.44	82.57
	HMM	86.67	100

Table 6: The statistics about the appearance of ODI detection.

Performance	Ground truth	Recall (%)	Precision (%)
Before refining	93	91.4	93.4
After refining	93	97.8	92.0

3.4. Multimodal structure analysis and event detection

Firstly, we show the experimental results of video analysis and event detection by using visual information only, and then we show the experimental results of event detection by using multimodal approach. Table 6 shows the results of ODI detection. The first row of the table shows the results of ODI detection using MACM and the second row of the table shows the results after applying the refining algorithm.

The ground truth, in Table 6, was defined as the actual number of ODIs that occurred including the captured and uncaptured ground truths. The results of potential event detection is shown in Table 7. From the table, we can see that arbitrary number of events have been detected and classified to correct classes. Table 8 shows the results of event detection using multimodal approach. In the table, the offensive foul and defensive foul are tabulated together and shown as "Foul." Comparing with Table 7, we can conclude that the accuracy of event detection is improved significantly by combining the visual with the audio information.

Table 7: The statistics about the appearance of potential event detection.

Events	Ground truth	Recall (%)	Precision (%)
Events before the ODI	85	87.0	92.5
Events without the ODI	29	76.6	76.7

Table 8: Results of event detection using the multimodal approach.

Performance	Ground truth	Recall (%)	Precision (%)
Foul	25	96.1	96.1
Shot at the basket	51	94.5	89.5

4. CONCLUSION

We have presented a novel semantic analysis and annotation approach by using multimodal analysis of video and audio information and tested in basketball videos. In shot and scene boundary detection, motion prediction information are used to detect scene boundaries. Moreover, motion features, describing the total motion, camera motion, and object motion, are utilized for scene classification. At the same time, our proposed HMM-based method for audio keysound detection outperforms the previous SVM-based method, especially for the excited commentator speech and excited audience sounds. This conforms to the fact that the HMM-based method effectively captures rich contextual information so as to improve different keysounds' separability. Experimental results have also demonstrated the effectiveness of event detection by using the combination of audio and visual information. Utilizing our method, we can generate a detailed description for video structure and detect an arbitrary number of events in a basketball game. The annotation information generated by the proposed method can be further combined for high-level video-content description and that information can subsequently be utilized to index, search, and retrieval of video contents.

REFERENCES

[1] Y. H. Gong, L. T. Sin, C. H. Chuan, H. Zhang, and M. Sakauchi, "Automatic parsing of TV soccer programs," in *Proceedings of International Conference on Multimedia Computing and Systems (ICMCS '95)*, pp. 167–174, Washington, DC, USA, May 1995.

[2] Y.-P. Tan, D. D. Saur, S. R. Kulkami, and P. J. Ramadge, "Rapid estimation of camera motion from compressed video with application to video annotation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 10, no. 1, pp. 133–146, 2000.

[3] P. Xu, L. Xie, S.-F. Chang, A. Divakaran, A. Vetro, and H. Sun, "Algorithms and system for segmentation and structure analysis in soccer video," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '01)*, pp. 721–724, Tokyo, Japan, August 2001.

[4] A. Ekin, A. M. Tekalp, and R. Mehrotra, "Automatic soccer video analysis and summarization," *IEEE Transactions on Image Processing*, vol. 12, no. 7, pp. 796–807, 2003.

[5] H. Lu and Y.-P. Tan, "Content-based sports video analysis and modeling," in *Proceedings of 7th International Conference on Control, Automation, Robotics and Vision (ICARCV '02)*, pp. 1198–1203, Singapore, December 2002.

[6] Y. Fu, A. Ekin, A. M. Tekalp, and R. Mehrotra, "Temporal segmentation of video objects for hierarchical object-based motion description," *IEEE Transactions on Image Processing*, vol. 11, no. 2, pp. 135–145, 2002.

[7] L.-Y. Duan, M. Xu, T.-S. Chua, Q. Tian, and C.-S. Xu, "A mid-level representation framework for semantic sports video analysis," in *Proceedings of 11th ACM International Conference on Multimedia*, pp. 33–44, Berkeley, Calif, USA, November 2003.

[8] M. Han, W. Hua, W. Xu, and Y. H. Gong, "An integrated baseball digest system using maximum entropy method," in *Proceedings of 10th ACM International Conference on Multimedia*, pp. 347–350, Juan les Pins, France, December 2002.

[9] S. Nepal, U. Srinivasan, and G. Reynolds, "Automatic detection of goal segments in basketball videos," in *Proceedings of 9th ACM International Conference on Multimedia*, vol. 9, pp. 261–269, Ottawa, Ontario, Canada, September 2001.

[10] M. Xu, L.-Y. Duan, C.-S. Xu, M. Kankanhalli, and Q. Tian, "Event detection in basketball video using multiple modalities," in *Proceedings of 4th International Conference on Information, Communications and Signal Processing and the 4th Pacific Rim Conference on Multimedia (ICICS-PCM '03)*, vol. 3, pp. 1526–1530, Singapore, December 2003.

[11] M. R. Naphade and T. S. Huang, "Semantic video indexing using a probabilistic framework," in *Proceedings of International Conference on Pattern Recognition (ICPR '00)*, vol. 3, pp. 3083–3088, Barcelona, Spain, September 2000.

[12] C. G. M. Snoek and M. Worring, "Multimedia event-based video indexing using time intervals," *IEEE Transactions on Multimedia*, vol. 7, no. 4, pp. 638–647, 2005.

[13] Y. Rui, A. Gupta, and A. Acero, "Automatically extracting highlights for TV baseball programs," in *Proceedings of 8th ACM International Conference on Multimedia*, pp. 105–115, Los Angeles, Calif, USA, October–November 2000.

[14] M. Xu, N. C. Maddage, C.-S. Xu, M. Kankanhalli, and Q. Tian, "Creating audio keywords for event detection in soccer video," in *Proceedings of IEEE International Conference on Multimedia and Expo (ICME '03)*, vol. 2, pp. 281–284, Baltimore, Md, USA, July 2003.

[15] L. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1993.

[16] J. Assfalg, M. Bertini, C. Colombo, A. Del Bimbo, and W. Nunziati, "Semantic annotation of soccer videos: automatic highlights identification," *Computer Vision and Image Understanding*, vol. 92, no. 2-3, pp. 285–305, 2003.

[17] H. Pan, P. van Beek, and M. I. Sezan, "Detection of slow-motion replay segments in sports video for highlights generation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '01)*, vol. 3, pp. 1649–1652, Salt Lake City, Utah, USA, May 2001.

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

[19] Z. Xiong, R. Radhakrishnan, A. Divakaran, and T. S. Huang, "Audio events detection based highlights extraction from baseball, golf and soccer games in a unified framework," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 5, pp. 632–635, Hong Kong, China, April 2003.

[20] J. Nam and A. Tewfik, "Combined audio and visual streams analysis for video sequence segmentation," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '97)*, vol. 4, pp. 2665–2668, Munich, Germany, April 1997.

[21] C. Saraceno and R. Leonardi, "Identification of story units in audio-visual sequences by joint audio and video processing," in *Proceedings of International Conference on Image Processing (ICIP '98)*, vol. 1, pp. 363–367, Chicago, Ill, USA, October 1998.

[22] H. Yi, D. Rajan, and L. T. Chia, "A unified approach to detection of shot boundaries and subshots in compressed video," in *Proceedings of International Conference on Image Processing (ICIP '03)*, vol. 2, pp. 1005–1008, Barcelona, Spain, September 2003.

[23] L. H. Siew, R. M. Hodgson, and E. J. Wood, "Texture measures for carpet wear assessment," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 1, pp. 92–105, 1988.

[24] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions System, Man, and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.

[25] C. Stiller and J. Konrad, "Estimating motion in image sequences," *IEEE Signal Processing Magazine*, vol. 16, no. 4, pp. 70–91, 1999.

[26] R. Szeliski, "Video mosaics for virtual environments," *IEEE Computer Graphics and Applications*, vol. 16, no. 2, pp. 22–30, 1996.

[27] S. Young, G. Evermann, D. Kershaw, et al., *The HTK Book (for HTK Version 3.1)*, Cambridge University Engineering Department, Cambridge, UK, December 2002.

[28] B. S. Manjunath, P. Salembier, and T. Sikora, *Introduction to MPEG-7*, John Wiley & Sons, New York, NY, USA, 2002.

Song Liu received the B.E. degree from the Department of Computer Science & Technology, Huazhong University of Science & Technology, China, in 2001. He is currently studying for the Ph.D. degree at the School of Computer Engineering, Nanyang Technological University, Singapore. His research interests include image/video processing, semantic interpretation for image/video content, and content-based image/video retrieval.

Min Xu is a Research Staff in the Center of Multimedia and Networking Technologies, School of Computer Engineering, Nanyang Technological University, Singapore. She received a B.E. degree from the University of Science and Technology of China majoring in automation in 2000 and an M.S. degree in computer science from the National University of Singapore in 2003. Her research interests include semantic multimedia computing, semantics modeling in multimedia data, and audio/video signal processing.

Haoran Yi received the B.S. degree in electrical and information engineering from Huazhong University of Science & Technology, Wuhan, China, in 2002. He is working for his Ph.D. degree in the School of Computer Engineering at Nanyang Technological University, Singapore, now. His research interests include content-based video analysis and representation, image understanding, and other issues on image and video technology.

Liang-Tien Chia received the B.S. and Ph.D. degrees from Loughborough University, in 1990 and 1994, respectively. He is the Director of the Centre of Multimedia and Network Technology and also an Associate Professor in the Division of Computer Communications, School of Computer Engineering, Nanyang Technological University, Singapore. His research interests include image/video processing, image/video coding, and multimedia adaptation/transmission. He has published over 50 research papers.

Deepu Rajan received the B.E. degree in electronics and communication engineering from Birla Institute of Technology, Ranchi, the M.S. degree in electrical engineering from Clemson University, and the Ph.D. degree from the Indian Institute of Technology, Bombay. From April 1992 until May 2002, he was a Lecturer in the Department of Electronics at the Cochin University of Science and Technology, India. Since June 2002, he has been an Assistant Professor in the School of Computer Engineering at Nanyang Technological University, Singapore. His research interests include image and video processing, computer vision, and neural networks.

International Journal of Biomedical Imaging

Special Issue on Mathematical Methods for Images and Surfaces

Call for Papers

"The Midwest Conference on Mathematical Methods for Images and Surfaces" was held in the Michigan State University on April 18-19. It created an excellent forum for researchers from engineering, biological, and mathematical sciences to exchange ideas and keep up with new developments. To further disseminate research findings presented and exchanged in the conference, The *International Journal of Biomedical Imaging* will publish a special issue entitled "Mathematical Methods for Images and Surfaces."

The scope of this special issue is the same as that of the conference. However, to better fit the scope of the journal, research findings relevant to biomedical science and technology are particularly welcome. Original papers and high-quality overviews on a wide range of topics in images and surfaces are solicited for this special issue. Topics of interest include, but are not limited to:

- Geometric flows, higher-order curvature flows, gradient flows for image, and surface analysis
- Mumford-Shah functional
- Level set methods and their applications
- Wavelets, frames, and multiresolution analysis
- Mathematical algorithms for images and surfaces
- Image edge detection, segmentation, pattern recognition, and video analysis and processing
- Computational methods for biomedical imaging
- Algorithms for bioluminescence imaging, fluorescent imaging, PET imaging, ultrasound imaging, MRI, and tomography
- Computational methods for anatomy
- Mathematical analysis of protein and membrane surfaces

The papers solicited for this special issue are not restricted to the contributions presented during the Conference. Submissions from other researchers which fit the scope of this special issue are also welcome.

Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://www.hindawi.com/journals/ijbi/guidelines.html. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/ according to the following timetable:

Manuscript Due	October 1, 2009
First Round of Reviews	January 1, 2010
Publication Date	April 1, 2010

Lead Guest Editor

Guowei Wei, Department of Mathematics and Department of Electrical and Computer Engineering, Michigan State University, MI 48824, USA; wei@math.msu.edu

Guest Editors

Lalita Udpa, Department of Electrical and Computer Engineering, Michigan State University, MI 48824, USA; udpal@egr.msu.edu

Yang Wang, Chair of Department of Mathematics, Michigan State University, MI 48824, USA; ywang@math.msu.edu

Shan Zhao, Department of Mathematics, University of Alabama, AL 35406, USA; szhao@bama.ua.edu

Hindawi Publishing Corporation
http://www.hindawi.com

EURASIP Journal on Advances in Signal Processing

Special Issue on Advanced Equalization Techniques for Wireless Communications

Call for Papers

With the introduction of personal communications services and digital packet data services, broadband wireless technology has experienced a significant upswing in recent years. To support the fast-growing wireless market, wireless research has to cope with formidable challenges that stem from wireless fading and multipath effects, finite-precision DSP, high signal dimension, and limited device size, to name a few. The goal is to design wireless devices that attain high data rate and high performance at low complexity. To achieve this goal, an essential step is channel equalization.

An ideal equalizer should achieve high performance, high rate, and low complexity. The tradeoffs among these three metrics are fundamental yet challenging in both theoretical analysis and hardware implementation. The aim of this special issue is to bring together the state-of-the-art research contributions that address advanced techniques in channel equalization for wireless communications. The guest editors seek high-quality papers on aspects of advanced channel equalization techniques, and value both theoretical and practical research contributions. Topics of interest include, but are not limited to:

- Low-complexity equalizers for wireless fading channels, including those that exploit sparsity
- Iterative equalization and decoding (turbo equalization)
- Time- and/or frequency-domain equalization for OFDM or single-carrier systems
- Equalization for rapidly time-varying channels
- Equalization for MIMO channels
- Equalization for multiuser systems
- Equalizers with finite-bit precision
- Equalization for cooperative relay systems
- Joint channel estimation and equalization

Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://www.hindawi.com/journals/asp/guidelines.html. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/ according to the following timetable:

Manuscript Due	October 1, 2009
First Round of Reviews	January 1, 2010
Publication Date	April 1, 2010

Lead Guest Editor

Xiaoli Ma, Georgia Institute of Technology, USA; xiaoli@ece.gatech.edu

Guest Editors

Tim Davidson, McMaster University, Canada; davidson@mcmaster.ca

Alex Gershman, Ruhr-Universität Bochum, Germany; gershman@nt.tu-darmstadt.de

Ananthram Swami, Army Research Lab, USA; a.swami@ieee.org

Cihan Tepedelenlioglu, Arizona State University, USA; cihan@asu.edu

Hindawi Publishing Corporation
http://www.hindawi.com

EURASIP Journal on Advances in Signal Processing

Special Issue on Advanced Image Processing for Defense and Security Applications

Call for Papers

The history of digital image processing can be traced back to the 1920s when digital images were transferred between London and New York. However, in the past, the cost of processing was very high because the imaging sensors and computational equipments were very expensive and had only limited functions. As a result, the development of digital image processing was limited.

As optics, imaging sensors, and computational technology advanced, image processing has become more commonly used in many different areas. Some areas of application of digital image processing include image enhancement for better human perception, image compression and transmission, as well as image representation for automatic machine perception.

Most notably, digital image processing has been widely deployed for defense and security applications such as small target detection and tracking, missile guidance, vehicle navigation, wide area surveillance, and automatic/aided target recognition. One goal for an image processing approach in defense and security applications is to reduce the workload of human analysts in order to cope with the ever increasing volume of image data that is being collected. A second, more challenging goal for image processing researchers is to develop algorithms and approaches that will significantly aid the development of fully autonomous systems capable of decisions and actions based on all sensor inputs.

The aim of this special issue is to bring together researchers designing or developing advanced image processing techniques/systems, with a particular emphasis on defense and security applications. Prospective papers should be unpublished and present innovative research work offering contributions either from a methodological or application point of view. Topics of interest include, but are not limited to:

- Multispectral/hyperspectral image processing for object tracking and classification with emphasis on defense-related targets and objects
- Real-time image processing for surveillance, reconnaissance, and homeland security
- Biometric image processing for personal authentication and identification with emphasis on homeland security applications
- Image encryption for secure image storage and transmission
- Image processing to enable autonomous and intelligent control for military, intelligence, and homeland security applications
- Image processing for mental workload evaluation with emphasis on homeland security applications
- Image interpolation and registration for object visualization, tracking, and/or classification

Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://www.hindawi.com/journals/asp/guidelines.html. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/ according to the following timetable:

Manuscript Due	December 1, 2009
First Round of Reviews	March 1, 2010
Publication Date	June 1, 2010

Lead Guest Editor

Yingzi (Eliza) Du, Department of Electrical and Computer Engineering, Indiana University-Purdue University Indianapolis, 723 W. Michigan Street, SL 160, Indainapolis, IN 46259, USA; yidu@iupui.edu

Guest Editors

Robert Ives, Department of Electrical Engineering, US Naval Academy, 105 Maryland Avenue, MS 14B, Annapolis, MD 21402, USA; ives@usna.edu

Alan van Nevel, Image and Signal Processing Branch Research Department, Naval Air Warfare Center, 1900 N Knox Road, M/S 6302 China Lake, CA 93555 USA; alan.vannevel@navy.mil

Jin-Hua She, School of Computer Science, Tokyo University of Technology, 1404-1 Katakura, Hachioji, Tokyo 192-0982, Japan; she@cs.teu.ac.jp

Hindawi Publishing Corporation
http://www.hindawi.com