IEEE Trans. Multimedia, vol. 8, no. 3, June 2006, pp. 585-599

FUSION OF AUDIO AND MOTION INFORMATION ON HMM-BASED HIGHLIGHT EXTRACTION FOR BASEBALL GAMES

*Chih-Chieh Cheng and Chiou-Ting Hsu**

Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan

Corresponding author: cthsu@cs.nthu.edu.tw

ABSTRACT

This paper aims to extract baseball game highlights based on audio-motion integrated cues. In order to better describe different audio and motion characteristics in baseball game highlights, we propose a novel representation method based on likelihood models. The proposed likelihood models measure the "likeliness" of low-level audio features and motion features to a set of predefined audio types and motion categories, respectively. Our experiments show that using the proposed likelihood representation is more robust than using low-level audio/motion features to extract the highlight. With the proposed likelihood models, we then construct an integrated feature representation by symmetrically fusing the audio and motion likelihood models. Finally, we employ Hidden Markov Model (HMM) to model and detect the transition of the integrated representation for highlight segments. A series of experiments have been conducted on a 12-hour video database to demonstrate the effectiveness of our proposed method and show that the proposed framework achieves promising results over a variety of baseball game sequences.

Keywords: highlight extraction, audio feature, camera motion, object motion intensity, likelihood model, Hidden Markov model

I. INTRODUCTION

With the ever-increasing entertaining video data, such as TV broadcasting, news, movies and documentaries, entirely manual annotation for video database is neither feasible nor appropriate nowadays. Video archiving technology has been developed to cope with the rapidly growing database and to provide more efficient service and convenience for end users. In recent years, sport videos have drawn increasing attention in automatic video analysis [1-12], since they are globally widespread and draw large audiences. In addition, as more convenient digital equipments emerge, it is much easier to record or further archive digital video data for general users at home. Moreover, the distribution of sport videos over internet further increase the necessities for automatic video analysis, such as quick browsing through a well-organized video, or detecting and recording some exciting highlights for later review. Nevertheless, automatic video analysis is still far from satisfactory in terms of semantic sense due to the large gap between semantics and feature interpretations of video data.

Video browsing technologies developed in recent years are generally divided into two categories: video summarization [13-15] and video skimming [16-18]. Video summarization shows a static capsule representation of a long video in terms of a set of key frames, whereas video skimming constitutes a dynamic representation that generates a shorter abstraction of a long video. Both of these technologies aim to provide users a fast browsing tool to snatch the main content of long video data in a relatively

short time. Two main issues in video browsing technologies include 1) how to extract important contents, and 2) how to distribute those important contents into limited display duration or space. One of the possible solutions to the first issue is highlight extraction. Highlight extraction can be viewed either as a separate application apart from the abovementioned two categories or as a pre-work of video skimming, since the extracted highlight captures most of the important or exciting contents in video and also provides a kind of dynamic video abstraction.

Highlight extraction has been extensively studied for sports videos [3-5, 9-11, 23, 29, 31]. In sport videos, where almost all the plays are rule-based and predictable, highlight extraction is more applicable to capture several specific "interesting events" from sport games, such as homeruns in baseball games and goal shooting in soccer games. With these well-defined structures, many domain-specific features and knowledge have been employed and incorporated into sport event detecting systems [1-3, 9, 22].

The main difficulty for highlight extraction lies in the semantic-level representation of what human being recognize as "interesting" events. The first step toward high-level content extraction is associated with content description, which is generally related to feature extraction and representation. Conventional image sequence processing typically relies on visual features only. Not except for video events analysis, most studies depend on low-level image features [1-3], such as dominant color ratio, edge descriptors, and object-based descriptors (e.g. players' height), which are mostly domain-specific. For example, in [2], the authors employed cinematic features, including dominant color ratio and shot boundary, and object-based features, such as the aspect ratio of referee region, to accomplish various events detection and summarization for soccer game. The detection process in [2] depends on several rule-based criteria which exhibit the commonality among various soccer games. Motion information is another visual feature that has drawn lots of attention presently. Motion features' temporal transition inherently represents the dynamics existing in videos and has been proved to be effective in semantic event detection [19-22]. Most important of all, motion features are capable to accommodate to environmental varieties for a certain sports events and retain better reliability through different conditions than other visual features. Global motion parameters such as pan and zoom have been adopted in [20]. Along with a finite state machine, the authors in [20] devoted to recognize several soccer events, including goals and penalties.

Compared to audio features, low-level visual features are relatively reliable even in noisy environment and easy to be analyzed through human eyes. However, visual features often capture very little semantic information of exciting video contents. For example, using visual features alone is unable to distinguish a close up of an audience from that of a player after scoring, since both cases have similar colors, shapes and camera motion. Hence, event detection systems based on audio analysis has flourished recently, since audio processing requires low computational complexity and performs better for semantic content extraction [10, 11]. In sport games, emotions of audience truthfully reflect the exciting degree of the play, through cheering, clapping, and excited shouting. Xiong et al. [11] concentrated on audio source only and developed an audio classification based approach. In [11], the authors identified applause/cheering segments and use the length of these segments to detect highlights. However, though audio features seem a good representation in semantic sense, their extraction is quite vulnerable to noisy conditions and thus the extraction results are highly related to several pre-defined heuristic assumptions. Moreover, highlight extraction based on audio classes is quite difficult to accommodate to environmental changes.

Consequently, several studies have proceeded to combine both audio and visual information existing in video streams to detect different events. Some of them concentrate on one type of information and use the other one as an auxiliary to improve the accuracy [12, 26-28]. In [12], the authors first use visual descriptors in conjunction with a controlled Markov chain (CMC) to classify each shot and select candidates for certain events. At the next stage, audio descriptors are incorporated to inspect each candidate to make the final decision. In order not to lose any benefit from either of the two types of features, several methods [29-33] have proposed to combine both types of features in a symmetric way.

For example, Dahyot et al. [32] proposed to calculate the probabilities of tennis events based on audio and visual feature separately, and combined both probabilities by a joint likelihood method.

The next step toward semantic content extraction is to define an appropriate measure for the target events. The measuring approaches adopted by most of the existing systems can be categorized into two general classes: deterministic reasoning and probabilistic inference [6]. The deterministic approaches generally depend on several domain-specific rules, related to various kinds of sport games [1-3, 7-12, 26-33]. Rui et al. [10] utilized audio features, including energy and pitch, and assumed that exciting segments are highly correlated with the announcer's excited speech. They measure the occurrence of both excited speech and ball-hit as highlights. In an extended work of [11], Xiong et al. [29] incorporated motion activity into their framework. They combined the classification labels based on audio and motion activity as joint histograms and utilized these histograms to detect interesting events. Although the abovementioned methods work well under several situations, the detection results are very sensitive to their predefined criteria, such as the predefined length of one cheering segment.

Many recent works have employed probabilistic model, such as Hidden Markov Model (HMM), to discover the temporal dependency of features [4-5, 23-25]. In [4], camera motion parameters are first quantized and modeled by a HMM, and then used to detect highlights in football games, such as penalty and free kick. However, as it is hard to design a suitable quantization level for motion descriptors, the number of observation symbols usually becomes very large. Chang et al. [23] assumed that most highlights in baseball games are composed of certain scene types and these scenes exhibit specific transition in time. They described each highlight event by a HMM in which hidden states are represented by their predefined scene types. However, in practice, some predefined scenes or images in interesting events may not appear in the assumed order or may never appear actually. On the other hand, interesting events may contain other scenes that do not belong to the predefined types, and a sequence of scenes that matches those predefined types may probably not an interesting event. Since automatic classification of video scenes into semantic types based on low-level audio-visual features are not robust enough to accommodate various situations and environments, highlight extraction according to the classification of scenes will inevitably bias the final result.

In this paper, we also employ a HMM to capture the temporal transition of baseball games but without predefining any highlight events or assuming any transition of scene types in advance. We aim to integrate both the audio and visual features symmetrically into a unified highlight extraction system. In order to improve the reliability of using audio-visual features to interpret the video semantics, we propose a novel feature representation method based on likelihood models. Our attempt here is to model the transition of time-varying audio and motion properties in the highlight segments and to avoid the extraction result's being biased by the problem of "hard classification". For example, if we directly classify each video frame into some specific audio and motion classes and represent the segment by its class labels, then the succeeding highlight training and detection processes will mostly rely on its class labels. Once a segment does not belong to any specific class or consists of multiple classes simultaneously, the method of "hard classification" will severely deteriorate the detection performance. For this reason, we propose using the likelihood models as a "soft" representation method for describing the transition of both audio and motion properties. After we measure the audio and motion likelihood representation from training data, we use a HMM to capture the temporal transition of the joint audio-visual likelihood vectors. Finally, at the detection stage, we measure the likelihoods of each video segment from the trained HMM to determine the final extraction results.

The rest of this paper is organized as follows. The audio-based and the motion-based feature representation are described in Section 2. In Section 3 we elaborate on how we integrate these two representation models with HMM to detect highlight segments. We demonstrate the performance and analyze the results of our work in Section 4, and conclude this paper in Section 5.

II. FEATURE EXTRACTION AND REPRESENTATION

In this section, we present the audio-visual feature extraction and our proposed representation method by likelihood models.

A. Audio Information

Audio source has its advantages of computational simplicity and remarkable representation of content semantics. Inspecting sequences of baseball games, we observe that exciting segments usually consist of several transitions between different audio types. Instead of classifying audio segments to certain predefined audio types, we propose to compute the likelihood of each audio segment to a set of audio types as our audio representation. We model each audio type as a Gaussian Mixture Model (GMM) of low-level audio features and estimate the model parameters from their corresponding training data. In the following subsections, we present the low-level audio feature extraction process as well as the conditional likelihood computation based on GMMs of the audio types.

1) Low-Level Audio Feature Extraction

We first divide each input audio signal into frames of 256 samples, with 128 samples overlap between adjacent frames. Each frame is multiplied by a hamming window function. A number of low-level audio features are then extracted from these audio frames [34, 35], including zero crossing rate (ZCR), pitch period and 12-Mel Frequency Cepstral Coefficients (MFCC). We briefly introduce each of the features as follows.

- *Zero Crossing Rate (ZCR):* ZCR, a commonly used temporal feature, counts the number of times that an audio signal $s_n(i), i = 0, \cdots, N-1$ crosses its zero axis. We define ZCR as

$$Z(n) = \frac{1}{2}\left(\sum_{i=1}^{N-1} \left| sign\left(s_n(i)\right) - sign\left(s_n(i-1)\right) \right| \right) \frac{f_s}{N}, \tag{1}$$

  where

$$sign[s_n(i)] = \begin{cases} 1, & if \quad s_n(i) >= 0 \\ -1 & if \quad s_n(i) < 0 \end{cases}, \tag{2}$$

  and f_s denotes the sampling rate.

- *Pitch Period:* Pitch is a popular audio feature, which measures the fundamental frequency of an audio signal. There are many kinds of approaches [36] to estimate pitch. We employ the autocorrelation function, $R_n(k)$, which is given by

$$R_n(k) = \frac{\sum_{i=0}^{N/2-1} s_n(i) s_n(i+k)}{\sum_{i=0}^{N/2-1} s_n(i+k)^2}, \tag{3}$$

  where $R_n(k)$ denotes the kth function value of the nth frame. We then detect periodic peaks in autocorrelation function to catch periods of repetitive signals inherited in harmonic sounds, such as music and voiced speech.

- *Mel Frequency Cepstral Coefficients (MFCC)*: These coefficients are computed from the magnitude spectrum of audio signals using the equation [38]

$$c_s(n) = \frac{1}{N'} \sum_{k=1}^{N'} \widetilde{Y}(k) \cos\left(k \frac{\pi}{N'} (s - 0.5) \right), \tag{4}$$

where $c_s(n)$ is the sth coefficients of the nth frame, and $\tilde{Y}(k)$ is the kth filter bank, which is the representative of the critical band in human auditory system. The first twelve coefficients $c_s(n), s = 1,2,\cdots,12$ are generally adopted in audio signal analysis.

2) Audio Representation using Likelihood Model

The 14 audio features (i.e. ZCR, pitch period, and 12 MFCCs) extracted from each audio frame constitute a 14-dimensional feature vector. However, the audio feature vector alone provides very limited information for distinguishing exciting segments, because the length of one frame (about 23ms) is usually too short to represent meaningful events and the transition of these feature vectors exhibits merely noise-like characteristic. Our attempt here is to measure the likelihood of a group of feature vectors (e.g., audio clip) belonging to a certain audio types. As discussed before, we did not classify each group of feature vectors into a definite class and analyze the transition of a sequence of class labels. Instead, we focus on how "likely" that a group of feature vectors belongs to each audio type and then model the transition of the likelihood for further process.

As suggested in other audio-based methods [10] [11], we define five audio types for baseball games: 1) ball hit, 2) cheering, 3) music, 4) speech, and 5) speech with music background. Among these audio types, the two types, music and speech with music background, usually appear in the commercial segments, while the other three types usually appear in running commentary and exciting events.

We model each audio type using a GMM to describe the distribution of its feature vectors. One of the advantages of GMM is its capability to describe complex distributions. Some audio data are diverse in their properties, such as music, which might consist of different instruments and singings. Moreover, GMM itself could inherently reflect the weighting of each feature from its corresponding covariance matrix, and hence it is more flexible for describing different audio types. For example, if a certain feature in one audio type is not representative enough, then the variance on this feature would become rather large and its dispersive distribution will automatically reflect the weighting of this feature.

Let π_{jr} denote the prior probability of the rth Gaussian component in the jth audio type. The likelihood of an audio feature vector $\mathbf{x}^a$ belonging to the jth audio type is defined by a GMM as follows:

$$p\left(\mathbf{x}^a \middle| \Psi_j\right) = \sum_{r=1}^{R_j} \pi_{jr} p\left(\mathbf{x}^a \middle| \Psi_{jr}\right), \tag{5}$$

where Ψ_j and Ψ_{jr} are model parameters for the jth audio type and its rth Gaussian component, respectively [39]. R_j denotes the number of Gaussian components in the jth audio type, $\mathbf{x}^a$ is the 14-dimensional audio feature vector, and $p\left(\mathbf{x}^a \middle| \Psi_{jr}\right)$ represents the rth Gaussian distribution with parameter set Ψ_{jr}.

For each audio type, we train the model parameters using EM algorithm through a manually classified training database. EM algorithm has been extensively used in many applications to train model parameters and provides a convenient way to maximize the log-likelihood over all training data,

$$\log L = \sum_{i=1}^{N} \log p\left(\mathbf{x}_i^a \middle| \Psi_{g_i}\right), \tag{6}$$

where g_i denotes the audio type to which the ith data belongs. Figure 1 shows the trained densities of two audio types, music and speech. Note that this illustration is a projection from 14-dimension into a 2-dimensional feature space and does not precisely reflect their actual overlap and distinction. Nevertheless, we can still get a picture of how these two types distribute in the 2-D space. The symbol "x" indicates the highest probabilities in the two densities, and although there are overlaps between these two types, we still can see that these two peaks are sufficiently separate from each other. Moreover, the model parameters learned from our training data show that the GMM of music type consists of 9 Gaussian

components while the GMM of speech type consists of 8 components. Hence, the density of music type spreads more widely than that of speech type does. These results confirm our intuition that the composition of music type is more complex than that of speech type.

Before calculating the likelihood value, we use a local energy threshold [40] and a global energy threshold to decide whether an audio frame and a clip contain only background noises or not, respectively. If the ratio of the number of noisy frames to total frames in one clip exceeds a certain amount, the likelihoods of that clip are set to zero.

One of the crucial problems in GMM training is how to decide the number of mixture components in a model. The most common solution is to select the number empirically for each type, but it may lead to overfitting or underfitting of the data. Some other model selection methods based on information theory have been proposed in the literature [41]. Minimum description length (MDL) is one of the alternatives and is adopted here as our model selection criterion. We select R_j that maximize

$$\log L_j - \frac{m_{R_j}}{2} \log N_j, \tag{7}$$

where L_j and N_j indicate the likelihood of data and the number of data in the jth audio type, and m_{Rj} denotes the number of free parameters needed for a GMM with R_j mixtures.

It is widely known that EM is highly dependent on initialization. We use multiple random starts and choose the estimate with the highest likelihood to initialize the EM algorithm. This initialization scheme has been proved to be effective [42]. The covariance matrix is initialized as 10% of the total covariance of training data, and only the diagonal elements are used as the model parameters for the sake of computational complexity. The training audio database, containing 665 clips, is collected from our baseball game database, and manually labeled in advance. All audio clips in this database are of 0.5 sec to 10 sec, with average duration around 5 sec, mono PCM files and sampled at 11025Hz.

After we obtain the model parameters of all the audio types from training data, we can now calculate the likelihoods of each audio clip to the five audio types using (5) and hence get one 5-dimensional likelihood vector to represent the audio clip.

B. Motion Information

Camera movements and object motion intensity in meaningful sport events generally follow a certain regulations in temporal domain. Specifically, in sport games, multiple cameras are fixed at several locations with simple motions, such as panning and zooming, for tracking players and the ball. The transition context of the camera movements and object motion intensity can hence be modeled by a HMM to extract highlight segments. In addition, in comparison with the conventional visual features, such as dominant color ratio or shape of players, motion information is less vulnerable to the change of different environments (e.g. different baseball fields or illumination change).

The motion features adopted in this work are composed of two parts: camera motion and object motion intensity. These features are directly derived from the motion vector fields (MVFs) of MPEG-1 compressed video streams. We compute these features on each P-frame. Since there may be different intervals between every pair of P-frames in video sequences, before extracting the MVFs from MPEG-1 streams, we normalize each motion vector with the length between the current frame and its reference frame.

1) Camera Motion and Its Likelihood Model

Since most camera motions attempt to track either the ball or runners, camera rotation seldom occurs in sport game sequences [4][23]. Hence, we disregard the rotation parameter from our model and represent camera motion using three parameters, including zooming, horizontal and vertical translations. Our camera motion is represented by

$$\mathbf{U}'_{x,y} = f_z \mathbf{U}_{x,y} + \Delta, \tag{8}$$

where $\mathbf{U}'_{x,y} = (x', y')^T$ is the position of $\mathbf{U}_{x,y} = (x, y)^T$ in the previous frame, f_z is the zoom factor and $\mathbf{\Delta} = (\Delta x, \Delta y)^T$ is the translational vector. We adopt the least-square method [44] to estimate the three motion parameters by minimizing

$$E(f_z, \mathbf{t}) = \sum_{x,y} \left\| \hat{\mathbf{U}}_{x,y} - \mathbf{U}'_{x,y} \right\|^2, \tag{9}$$

where $\hat{\mathbf{U}}_{x,y}$ is the predicted position of $\mathbf{U}_{x,y}$ in the previous frame. To reduce the disturbance from moving objects, we redefine the minimization process as

$$\min_{f_z,\mathbf{t}} E(f_z, \mathbf{t}) = \min_{f_z,\mathbf{t}} \sum_{\left\| \hat{\mathbf{U}}_{x,y} - \mathbf{U}'_{x,y} \right\| < T} \left\| \hat{\mathbf{U}}_{x,y} - \mathbf{U}'_{x,y} \right\|^2, \tag{10}$$

In (10), when a predicted position significantly differs from the actual position, the motion of this position may probably contain object motion and will be eliminated from our minimization process.

There are still several difficulties with this estimation approach. First, the motion vectors extracted from MPEG-1 streams are usually noisy and do not represent the "real motion" of each macroblock. That is because motion estimation in MPEG standard just aims to reduce the matching residual and to achieve a higher compression ratio. The noisy motion vectors will then result in unreliable estimation of camera motion. In addition, another kind of artificial noise may also exist due to tiny camera vibrations which often occurs during tracking the ball or any other targets in the play. The camera vibrations will also result in unstable estimation of camera motion and make the transitions of camera motion noise-like.

In the following, we propose a likelihood representation to overcome the abovementioned difficulties. Instead of measuring "real quantity" of camera motion, we measure the likelihood of a video frame under three camera motion categories, including panning, zooming and tilt, which are denoted by Ψ_{pan}, Ψ_{zoom}, and Ψ_{tilt}, respectively. This kind of soft representation shows how "likely" a certain camera motion category appears in each video frame and can effectively avoid the instability and unreliability presented in the camera motion estimation. Moreover, a scene often contains various kinds of camera motions simultaneously, such as panning with zooming, and has complicated motion characteristics. It is quite difficult to classify one frame to a predefined class of camera motions. Therefore, the proposed likelihood model attempts to represent how confident one camera motion exists in one video frame and to exhibit the composition of camera motions within the scene.

Here we outline the derivation for the likelihood model. Let $\mathbf{\Delta}_t = (\Delta x_t, \Delta y_t)$ and $f_{z,t}$ denote the estimated camera motion parameters (i.e. horizontal translation, vertical translation and zooming factor) at time t, and $\mathbf{M}_t$ represent the set of motion vectors. We define the set of residual motion vectors $\mathbf{M}_t^{res}$ as the difference between the actual position and the position predicted using the estimated camera motion, that is,

$$\mathbf{M}_t^{res} = \left\{ \mathbf{U}'_{x,y} - \hat{\mathbf{U}}_{x,y} \middle| \forall x, y \right\}, \tag{11}$$

which is illustrated in Fig. 2. Given a specific motion category, the goal is to evaluate the likelihood of the observed camera motion parameters to this category using a class-conditional probability density function. In order to make the derivation more tractable, we assumed that the occurrence of a set of residual motion vectors $\mathbf{M}_t^{res}$ is independent of motion categories. Especially in the case of complex global motion mixed with object motion, the residual motion vectors $\mathbf{M}_t^{res}$ usually appear to be chaotic and could be treated as independent of any camera motion category.

Taking panning as an example, given the motion category Ψ_{pan}, our goal is to evaluate the joint likelihood $p(\Delta x_t, \mathbf{M}_t^{res}, \Delta x_{t-1} \mid \Psi_{pan})$ of observing Δx_t and $\mathbf{M}_t^{res}$ at time t and Δx_{t-1} at time t-1. Under the assumption that the set of residual motion vectors $\mathbf{M}_t^{res}$ and the motion category Ψ_{pan} are independent (i.e. $p(\mathbf{M}_t^{res} \mid \Psi_{pan}) = p(\mathbf{M}_t^{res})$) and the prior probabilities $p(\mathbf{M}_t^{res})$, $p(\Psi_{pan})$ and $p(\Delta x_t)$ are constants, the joint likelihood is factorized as

$$\begin{aligned} p(\mathbf{x}_{pan}^{v} \mid \Psi_{pan}) &= p(\Delta x_t, \mathbf{M}_t^{res}, \Delta x_{t-1} \mid \Psi_{pan}) \\ &= p(\Delta x_t, \Delta x_{t-1} \mid \Psi_{pan}, \mathbf{M}_t^{res}) p(\mathbf{M}_t^{res} \mid \Psi_{pan}) \\ &= p(\Delta x_t \mid \Psi_{pan}, \mathbf{M}_t^{res}, \Delta x_{t-1}) p(\Delta x_{t-1} \mid \mathbf{M}_t^{res}, \Psi_{pan}) p(\mathbf{M}_t^{res} \mid \Psi_{pan}) \\ &\propto p(\Delta x_t \mid \Psi_{pan}, \mathbf{M}_t^{res}, \Delta x_{t-1}) p(\Delta x_{t-1} \mid \Psi_{pan}) \end{aligned} \tag{12}$$

where $\mathbf{x}_{pan}^{v}$ denotes the observed camera motion parameters for panning, and the term $p(\Delta x_{t-1} \mid \mathbf{M}_t^{res}, \Psi_{pan}) = p(\Delta x_{t-1} \mid \Psi_{pan})$ because the translation Δx_{t-1} at time t-1 is independent of the residual motion vectors $\mathbf{M}_t^{res}$ at time t. The term $p(\Delta x_t \mid \Psi_{pan}, \mathbf{M}_t^{res}, \Delta x_{t-1})$ in the last equation of (12) can be further factorized using Bayes formula as

$$\begin{aligned} & p(\Delta x_t \mid \Psi_{pan}, \mathbf{M}_t^{res}, \Delta x_{t-1}) \\ &= \frac{p(\Psi_{pan}, \mathbf{M}_t^{res}, \Delta x_{t-1} \mid \Delta x_t) p(\Delta x_t)}{p(\Psi_{pan}, \mathbf{M}_t^{res}, \Delta x_{t-1})} \\ &= \frac{p(\Psi_{pan} \mid \Delta x_t, \Delta x_{t-1}, \mathbf{M}_t^{res}) p(\mathbf{M}_t^{res} \mid \Delta x_t, \Delta x_{t-1}) p(\Delta x_{t-1} \mid \Delta x_t) p(\Delta x_t)}{p(\Delta x_{t-1}, \mathbf{M}_t^{res} \mid \Psi_{pan}) p(\Psi_{pan})} \\ &= \frac{\frac{p(\Delta x_t, \Delta x_{t-1} \mid \Psi_{pan}) p(\Psi_{pan})}{p(\Delta x_t, \Delta x_{t-1})} \frac{p(\Delta x_t \mid \mathbf{M}_t^{res}) p(\mathbf{M}_t^{res})}{p(\Delta x_t)} \frac{p(\Delta x_t \mid \Delta x_{t-1}) p(\Delta x_{t-1})}{p(\Delta x_t)} p(\Delta x_t)}{p(\Delta x_{t-1} \mid \Psi_{pan}, \mathbf{M}_t^{res}) p(\mathbf{M}_t^{res} \mid \Psi_{pan}) p(\Psi_{pan})} \\ &\propto \frac{\frac{p(\Delta x_t, \Delta x_{t-1} \mid \Psi_{pan})}{p(\Delta x_t \mid \Delta x_{t-1})} p(\Delta x_t \mid \mathbf{M}_t^{res}) p(\Delta x_t \mid \Delta x_{t-1})}{p(\Delta x_{t-1} \mid \Psi_{pan})} \\ &\propto \frac{p(\Delta x_t, \Delta x_{t-1} \mid \Psi_{pan}) p(\Delta x_t \mid \mathbf{M}_t^{res})}{p(\Delta x_{t-1} \mid \Psi_{pan})} \end{aligned} \tag{13}$$

Hence, we get

$$\begin{aligned} p(\mathbf{x}_{pan}^{v} \mid \Psi_{pan}) &\propto p(\Delta x_t, \Delta x_{t-1} \mid \Psi_{pan}) p(\Delta x_t \mid \mathbf{M}_t^{res}) \\ &= p(\Delta x_t \mid \Delta x_{t-1}, \Psi_{pan}) p(\Delta x_{t-1} \mid \Psi_{pan}) p(\Delta x_t \mid \mathbf{M}_t^{res}) \end{aligned} . \tag{14}$$

Next, we further assume the following relations to model the camera pan:

$$p(\Delta x_t \mid \mathbf{M}_t^{res}) = N(E(f_{z,t}, \mathbf{\Delta}_t) \mid 0, \eta_E) / \Lambda_E , \tag{15}$$

$$p(\Delta x_t \mid \Delta x_{t-1}, \Psi_{pan}) = \exp\left[- \| \Delta x_t - \Delta x_{t-1} \|^2 \right] / \Lambda_{pan} , \tag{16}$$

$$p(\Delta x_{t-1} \mid \Psi_{pan}) = - N(\Delta x_{t-1} \mid 0, \eta_{pan}) / \Lambda'_{pan} + 1 , \tag{17}$$

where $E(f_{z,t}, \mathbf{\Delta}_t)$ denotes the minimized error function (or the energy of residual motion) calculated in (10), and Λ_E , Λ_{pan} and Λ'_{pan} are normalization factors. In (15) and (17), $N(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\eta})$ denote a Gaussian distribution over the random vector $\mathbf{x}$ with mean $\boldsymbol{\mu}$ and inverse covariance matrix $\boldsymbol{\eta}$:

$$N(\mathbf{x} \mid \boldsymbol{\mu}, \boldsymbol{\eta}) = |\boldsymbol{\eta}/2\pi|^{1/2} \exp\left[-\frac{1}{2}(\mathbf{x}-\boldsymbol{\mu})^T \boldsymbol{\eta}(\mathbf{x}-\boldsymbol{\mu})\right]. \tag{18}$$

Note that, the variance parameters in (15) and (17) are learned from our training data. Since the functions involved in (15) and (17) have only one dimension, it is not difficult to estimate the variance parameters for characterizing these distributions.

Equation (15) accounts for the confidence degree of the camera motion estimation. We assume that the larger the final estimation error (or the energy of residual motion) is, the less confidence we have for the estimated parameters. This model tackles the problem of unreliability presented in MPEG motion vector fields. Equation (16) assumes that, when a certain camera motion category exists in a video, the camera motion between two adjacent frames should be consistent. Hence, if two motion parameters (i.e. Δx_t and Δx_{t-1}) in adjacent frames are closer, we can conclude that a certain motion category (i.e. pan) is more probable to appear, and the problem of instability is henceforth tackled. In (17), the term $p(\Delta x_{t-1} \mid \Psi_{pan})$ accounts for a confidence level of the estimated camera motion parameter. In other words, this term correlates the likelihood between adjacent frames; moreover, it is intuitive that if the previous frame is more likely to be of a certain category of camera motion, so is the current frame.

Similar derivations can be obtained for both tilt and zoom. For tilt and zoom, the features used to calculate (14) are denoted as $\mathbf{x}^v_{tilt}$ and $\mathbf{x}^v_{zoom}$. The two assumptions adopted in (15)-(16) remain the same except that the panning parameter is replaced by tilt and zoom parameters. The assumption adopted in (17) is slightly different for the cases of tilt and zoom. In the case of camera tilt, the tilt parameter, variance and tilt motion pattern are substituted for those of panning in (17). On the other hand, in the case of camera zoom, we employ the strength of zooming effect instead of the magnitude of zooming parameter to measure the confidence level of the estimated zooming factor by

$$p(f_{z,t-1} \mid \Psi_{zoom}) = -N(s \mid 1, \eta_{zoom})/\Lambda_{zoom} + 1, \qquad \text{where } s = \begin{cases} f_{z,t-1}, & \text{if } f_{z,t-1} \geq 1 \\ 1/f_{z,t-1}, & \text{if } f_{z,t-1} < 1 \end{cases}. \tag{19}$$

As can be seen from the above formulations (12)-(19), we join the camera motion estimation, the accuracy related to the estimation phase, and the temporal dependencies of camera motion parameters altogether to compensate the instability and unreliability presented in traditional camera motion estimation. Here, we illustrate the proposed motion representation method with an example. Fig. 3 shows the resulted likelihood curve of horizontal translation (panning) for a one-minute homerun segment in a baseball game. Fig. 3 (a) shows the keyframes and descriptions for 8 segments, Fig. 3 (d) represents the quantity of horizontal displacement obtained from the minimization process defined in (10), and Fig. 3 (b) shows the likelihood estimated using our proposed method defined in (14). Fig. 3 (c) shows the manually labeled ground truth, where 1 and 0 indicate “pan” and “no pan” respectively. Note that the likelihood model is more reliable than the estimated parameters in the sense that it can promisingly reflect real camera motion category of the play. The estimated displacement exhibits noise-like attributes, especially during segments with complicated camera motion.

2) Object Motion Intensity and Its Likelihood Model

We assume that the object motion can be observed from the relative difference between real block motion and camera motion. Hence, without time-consuming camera motion compensation, we estimate the object motion intensity from the residual motion vectors $\mathbf{M}_t^{res}$ which resulted from subtracting the estimated

camera motion from the original motion vectors. Given $\mathbf{U}'_{x,y}$ as the true position of a pixel (x, y) in the previous frame, and $\hat{\mathbf{U}}_{x,y}$ as the estimated position of the pixel (x, y) in the previous frame according to the camera motion parameters, the residual motion intensity $\left\|\mathbf{M}_t^{res}(x,y)\right\|$ at the pixel (x, y) is measured by

$$\left\|\mathbf{M}_t^{res}(x,y)\right\| = \left\|\mathbf{U}'_{x,y} - \hat{\mathbf{U}}_{x,y}\right\|. \tag{20}$$

That is, if a certain block motion does not agree with the camera motion, it is more probable that a locally moving object exists in this block, as illustrated in Fig. 2.

Next, we use these residual motion intensities to measure the average residual motion intensity at time t:

$$Q_t = \frac{\sum_{x,y}\left\|\mathbf{M}_t^{res}(x,y)\right\| \times w(x,y)}{\sum_{x,y\in R} w(x,y)}, \tag{21}$$

where $w(x,y)$ is the weight for different spatial positions. With the object motion intensities $Q_{t\text{-}1}$ and Q_t and the residual motion vectors $\mathbf{M}_t^{res}$, the derivation of the likelihood model for the motion category Ψ_{obj} is derived as follows:

$$\begin{aligned} p\left(\mathbf{x}_{obj}^{v}\middle|\Psi_{obj}\right) &= p\left(Q_t, \mathbf{M}_t^{res}, Q_{t-1}\middle|\Psi_{obj}\right) \\ &= p\left(Q_t\middle|\Psi_{obj}, \mathbf{M}_t^{res}, Q_{t-1}\right)p\left(Q_{t-1}\middle|\Psi_{obj}, \mathbf{M}_t^{res}\right)p\left(\mathbf{M}_t^{res}\right) \\ &\propto p\left(Q_t\middle|\Psi_{obj}, \mathbf{M}_t^{res}, Q_{t-1}\right)p\left(Q_{t-1}\middle|\Psi_{obj}\right), \end{aligned} \tag{22}$$

where the first term can be similarly derived as in Eq. (13)

$$p\left(Q_t\middle|\Psi_{obj}, \mathbf{M}_t^{res}, Q_{t-1}\right) \propto \frac{p\left(Q_t, Q_{t-1}\middle|\Psi_{obj}\right)p\left(Q_t\middle|\mathbf{M}_t^{res}\right)}{p\left(Q_{t-1}\middle|\Psi_{obj}\right)}, \tag{23}$$

and hence

$$\begin{aligned} p\left(\mathbf{x}_{obj}^{v}\middle|\Psi_{obj}\right) &\propto p\left(Q_t, Q_{t-1}\middle|\Psi_{obj}\right)p\left(Q_t\middle|\mathbf{M}_t^{res}\right) \\ &= p\left(Q_t\middle|Q_{t-1}, \Psi_{obj}\right)p\left(Q_{t-1}\middle|\Psi_{obj}\right)p\left(Q_t\middle|\mathbf{M}_t^{res}\right) \end{aligned}. \tag{24}$$

Once again, we model the dependencies of the observed variables as

$$p\left(Q_t\middle|\mathbf{M}_t^{res}\right) = N\left(E\left(f_{z,t}, \mathbf{\Lambda}_t\right)\middle|\, 0, \eta_E\right)\Big/\Lambda_E\,, \tag{25}$$

$$p\left(Q_t\middle|Q_{t-1}, \Psi_{obj}\right) = \exp\left[-\left\|Q_t - Q_{t-1}\right\|^2\right]\Big/\Lambda_{obj}\,. \tag{26}$$

$$p\left(Q_{t-1}\middle|\Psi_{obj}\right) = -N\left(Q_{t-1}\middle|0, \eta_{obj}\right)\Big/\Lambda'_{obj} + 1\,. \tag{27}$$

Similarly, the variance parameters in (25) and (27) are learned from our training data.

The term in (25) again accounts for the estimation correction. Since we calculate the object motion intensity by removing the camera motion from the original motion and averaging the residual motion intensities, the correction of this estimation can hence represent the confidence degree of the camera motion estimation. Equations (26) and (27) are almost identical to those in the previous subsection, which stand for temporal consistency and strength of object motion, respectively.

Fig. 4 shows the resulted likelihood curve of object motion intensity for the same sequence in Fig. 3. Fig. 4 (b) is the ground truth labeled manually, and Fig. 4 (c) denotes the object motion intensity calculated by (21). Notice that since we incorporate the confidence level of camera motion estimation into

the likelihood computation, the likelihood becomes more reliable and comes closer to the ground truth than the original object motion intensity.

III. HIGHLIGHT EXTRACTION WITH INTEGRATED INFORMATION BASED ON HMM

The idea behind the proposed joint audio-visual HMM model relies on two observations. The first is that baseball highlights generally contain a certain temporal transition of basic audio types, regular camera movements and object motion intensity. The most typical highlight in baseball game is homerun, in which temporal transition of audio types, camera motion and object motion is very distinctive. Our second observation is that the temporal transition in most exciting events is similar to that in homerun segments. As a result, our highlight extraction aims to detect exciting events such as homerun, spectacular slug and score, which exhibit similar transition patterns as homerun but with shorter length of exciting portion. Fig. 5 shows the overview of the proposed framework. In the following subsections, we introduce the integration of audio and visual models and the highlight extraction process based on HMM.

A. Fusion of Audio and Motion Information

Joint audio-visual highlight extraction technique has been developed in several literatures. Low-level visual features have their limitations to express high-level semantic meanings of scenes while audio signals can generally provide more semantic information, such as cheering of audience. On the other hand, the noise prevailing in audio signals is comparably high whereas motion information is more feasible against environmental noises. Thus, combination of both features is more capable to complement each other and to improve the reliability of the highlight extraction.

Most of the previous works focused on either audio-centric or visual-centric framework. Few of them have mentioned integrating these two frameworks symmetrically. The method we propose in this paper aims to combine these two frameworks without sacrificing benefits from either one and to capture the transitions of both audio and motion information simultaneously into one unified model.

By concatenating the proposed audio and motion likelihoods discussed in Sec. 2, we obtain a 9-dimensional likelihood vector as an input to the HMM:

$$\mathbf{O} = \left\{ \left(p\left(\mathbf{x}_i^v \middle| \Psi_i\right), p\left(\mathbf{x}^a \middle| \Psi_j\right) \right) \middle| i = \{\text{pan, tilt, zoom, object}\}, j = \{\text{ball hit, cheering, music, speech, speech with music nackground}\} \right\}, \quad (28)$$

where $\mathbf{x}^a$ denotes the 14-dimensional low-level audio feature vector, and $\mathbf{x}_i^v$ denotes the set of motion parameters used for calculating the likelihood of the motion category i, as defined in (12). Note that, the synchronization between audio and video signals must be carefully handled, since audio signals requires much higher sampling rate, generally 11kHz, while video signals is sampled at a slower frequency of 30 frames/sec. To solve the problem, we group audio feature vectors into audio clips, corresponding to the interval between successive P-frames, and construct one likelihood vector for each audio clip. Hence, we obtain a 9-dimensional likelihood vector for each P-frame.

B. Highlight Extraction using HMM

HMM has been applied to many areas as a powerful method to represent particular types of stochastic process [48]. We train the continuous observation HMM according to Baum-Welch algorithm. Conventional implementation issues in HMM include 1) number of states, 2) initialization, and 3) distribution of observation at each state. We give the detailed description of each essential element below.

- *State* **S**: unknown states of the given highlight segments. We select the number of states experimentally and adopt a four-state model in this work. Furthermore, since conventional joint audio-visual classification based on low-level audio-visual features is neither appropriate nor robust to adapt to different situations, we do not attempt to define each hidden state of the HMM to be a specified scene class.

- *Observation* **O**: the 9-dimensional likelihood vector of each P-frame.
- *Observation distribution* $p(\mathbf{O}|\mathbf{S})$: We model the observation distribution of state S as a GMM, which can be trained by EM algorithm or K-means algorithm, and the number of mixture components is decided by running several models and choosing the model with the highest likelihood. Experiments show that a single Gaussian distribution at each state is preferred.
- *Transition probability* **A**: the state transition probability, which can be learned from training data.
- *Initial state probability* $\mathbf{\Pi}$: the probability of occurrence of the first state, which is randomly initialized.

For highlight detection, we group likelihood vectors of each P-frame to get a one-minute segment with 0.5 minutes overlapping between adjacent segments and pass each segment to the detection process. If the likelihood of one segment is at the local maxima and higher than a certain threshold, we declare this segment as a highlight event.

However, our experiment shows that several uninteresting segments in a baseball game video (e.g., commercials) may also obtain a high likelihood in our HMM detection process. Hence, some post-processing schemes are needed to trim out these segments. Observing the commercial segments, we found that their shot cut rate is generally higher than that of other important segments. Hence, we further measure the shot-cut rate [45-47] for each one-minute segment and will include only those segments with shot cut rate less than a certain threshold into our detection process.

IV. EXPERIMENTAL RESULTS

Our experimental database comprises 24 baseball games recorded from Major League Baseball (MLB) TV broadcasts with total length more than 12 hours. The whole database is composed of MPEG-1 video streams in 352 × 240 resolution at 30 fps (or 29.97 fps to be exact). Among them, 40% are used as training database, and the rest as our test database. We manually select several representative homerun segments from the training database for training the proposed HMM. An audio database of 665 sound clips is also extracted from the training database as the training data for constructing the prototypical audio model.

The goal of this section is to show and analyze the effectiveness of the proposed highlight extraction framework with 1) fusion of both audio and visual information in video data, and 2) incorporation of temporal dependency of audio/visual information by HMM.

A. Ground Truth and Evaluation

The ground truth is established based on human evaluation of exciting segments in our database. This type of ground truth is common in video browsing technologies and video analysis [49, 15]. Although different people may have distinctive judgments about highlight events, there are common highlight segments existing in baseball game sequences that really excite the audience.

Let GT denote the number of highlights segments in the ground truth, *detected highlights* (D) indicate the number of highlights declared by our proposed framework, and *correct highlights* (C) denote the number of correctly detected highlights. Accordingly, *false positive* (FP) and *false negative* (FN) in the results are defined as

$$FP = D - C, \qquad FN = GT - C\,. \tag{29}$$

The overall evaluation is measured by accumulating of the numbers of detected highlights, correctly detected highlights, and ground truth of the whole test data.

B. Results of Audio-Visual Integrated Framework

In this experiment, we aim to demonstrate the effectiveness of our proposed integrated framework compared to using only audio feature (this method is referred to as the audio-based method) or motion feature (i.e. the motion-based method) for highlight extraction. In order to show a fair comparison on using audio-only, visual-only and audio-visual features, we adopt the same audio and motion feature representations described in Section 2 and use the same HMM training and detection processes as described in Section 3. The results of audio-based method, motion-based methods and our proposed integrated framework performed on test data set are shown in the left, middle and right columns of Table 1, respectively.

As can be seen from Table 1, the proposed integrated framework outperforms the other two methods on almost all the test sequences in terms of both the numbers of false positives and false negatives. This experimental result also clarifies our statements that the noises (e.g. disturbance from audience) prevailing in audio signal are comparably higher than that in motion information. On the other hand, even motion features are more stable, motion information alone is incapable of distinguishing true highlight segments. As a result, the false positives are much higher in both audio-based and motion-based frameworks, and combination of these two features indeed effectively improves the performance. Though the numbers of false positives are generally higher than the numbers of false negatives in all three cases, it is a tendency in event detection system to over-extract possible segments that match their predefined criteria rather than to minimize the number of extracted segments. In other words, while keeping the false positives in a reasonable range, our attempt is to minimize the amount of false negatives to achieve an optimal result.

Figure 6 shows the extraction results of test sequence T_0. The dotted line indicates the resulting likelihood curve from our proposed HMM framework with integrated audio and motion features, and the dash line denotes the detected highlights. Note that the post-process mentioned in Section 3.2 filters out some of the peaks that exceed the threshold. As shown in Fig. 6, the proposed methodology actually provides quite promising results. Figure 7 shows the extracted highlight segments with several keyframes to indicate each highlight. The first four rows correspond to the four correctly detected highlights in Fig. 6, and the last row shows one of the falsely detected highlights, which is also a representative of the majority of detection errors, since almost all the false positives are constituted of these kinds of shots. More details about these wrongly detected highlights will be addressed in the discussion section. Moreover, as the evaluation of the HMM likelihood is performed for every one-minute segment, with 0.5 minute overlapping, the extracted highlight sequences may not perfectly or exactly match the entire duration of a highlight, or may contain other non-highlight action, especially at the beginning and the end. However, as long as a segment contains any highlight event, the proposed framework has its capability of detecting the transition of audio and motion information in the highlight, regardless of how long the highlight may hold.

C. Trade-off between False Positives and False Negatives

The false positive values shown in Table 1 could be further reduced if additional post-processes are adopted. When observing the falsely detected highlights (which will be discussed later in Section 4.6), we find that most of them consist of a large portion of static frames. One possibility is to use the ratio of the number of static frames within each segment as a criterion. If this ratio of a detected highlight is higher than a certain threshold, we exclude this highlight segment from the final result. However, it is a tradeoff between the values of false positive and false negative. While we try to filter out more falsely detected highlights, some correctly detected highlights might also be discarded as well. The resulting FP and FN curve is shown in Fig. 8. This curve is a plot of the number of FP and FN versus the number of detected highlights obtained using different parameters (which is the threshold of static-frame ratio in our case), where lower threshold means to filter out more detected highlights. As can be seen from Fig. 8, it is very difficult to determine an optimal result to fit all applications. Further analysis on reducing both FP and FN values will be our major future work.

D. Probabilistic Inferring v.s. Deterministic Reasoning

In this subsection we compare the proposed HMM framework with previous researches based on deterministic detection process [10]. In deterministic detection process, highlight segments are generally assumed to have cheering from audience and intense camera motion due to ball- or other target-tracking. Hence, we implement the deterministic detection as detecting the segments with high cheering and large camera motion. Thus, the resulting likelihood of the deterministic detection is defined as

$$p(\mathbf{x}|\text{highlight}) = W_c\, p\left(\mathbf{x}^a \middle| \Psi_{cheering}\right) + W_p\, p\left(\mathbf{x}^v_{pan} \middle| \Psi_{pan}\right) + W_t\, p\left(\mathbf{x}^v_{tilt} \middle| \Psi_{tilt}\right) + W_z\, p\left(\mathbf{x}^v_{zoom} \middle| \Psi_{zoom}\right), \tag{30}$$

where $p\left(\mathbf{x}^v_{pan} \middle| \Psi_{pan}\right)$, $p\left(\mathbf{x}^v_{tilt} \middle| \Psi_{tilt}\right)$ and $p\left(\mathbf{x}^v_{zoom} \middle| \Psi_{zoom}\right)$ are defined according to (12), and $p\left(\mathbf{x}^a \middle| \Psi_{cheering}\right)$ is defined in (5). W_c, W_p, W_t and W_z are scaling factors used for normalizing different scales between audio and motion representations. Table 2 shows the detailed results for comparison performed on test data set. Again, in order to have a fair comparison and to exclude any impacts from audio-visual features or other issues, we use the same audio and motion representations proposed in Section 2 to implement the deterministic method.

In Table 2, the proposed HMM-based framework once again outperforms the deterministic method. Figure 9 is the detection result using deterministic method with test sequence T_0, and this result demonstrates that it is quite difficult to judge highlight segments based on the deterministic method, since occurrence of likelihood peaks is very noisy. The decision process used here is that for every 0.5 minute, if a peak surpassing a certain threshold exists in the subsequent one-minute segment, we consider this peak as a highlight. If more than one peak appear in one segment, we select the highest peak as a representative. Notice that, although the results are not as satisfactory as the HMM-based detection approach, with the proposed feature representation, it is still successful to extract those segments that match our assumption.

E. Comparison between Symmetric Combination and Visual-Centric Framework

Here, we compare the proposed symmetric audio-visual combination framework with the visual-centric method, which is further refined using audio information. The motion-based method shown in the leftmost column of Table 3 is the same as in Table 1. In this experiment, we consider that if the number of "cheering" clips in one segment exceeds a certain threshold, this segment is more probable to attract the audience and reflects higher degree of excitement. Hence, in addition to motion-based analysis, we perform the audio refinement by filtering the highlight segments extracted by the motion-based method with the portion of cheering clips in those segments. As can be seen from Table 3, our proposed symmetric framework again obtains a better result. Notice that although the audio refinement reduces the false positives in the motion-based method, it also increases the false negatives. That is because the portion of cheering in the highlight segments is not always so large or lasts so long, and the audio signals may have composite characteristics (e.g. speech with cheering) that make it more difficult to analyze cheering portion, as the aforementioned "hard classification" problem. Our proposed HMM-based framework with likelihood representation overcomes this difficulty since we capture the transition of likelihood information without considering the absolute length or portion of cheering in the highlight segments.

F. Discussion

A typical erroneous detection of highlights is observed from the last row in Fig. 7. All of the shots constituting the segment are almost static in motion and with pure speech of reporters. The shots of pitch view consist of only the motion of the pitcher, and the close-up shots generally capture the expression of a certain person, such as the coach or any players in the field. These shots dominate the main portion of a baseball game, including highlight segments as well. In our training sequences (homerun sequences), these kinds of shots mostly occur at and dominate the beginning and the end. Moreover, they are very easy to be modeled by a Gaussian distribution, since the composition of the static shots is simple, and all the features describing static scenes are quite concentrated. Consequently, one of the states in our HMM

model becomes to represent these static shots through the automatic learning process. If one segment contains many static shots as shown in Fig. 7, its resulting likelihood is very likely to be generated from this state of static shots. One of the possible solutions is to filter out the extracted highlights with little motion intensity. Nevertheless, such filtering will again result in a tradeoff between false positives and false negatives. Other possibilities, such as constraining state transition in HMM, will be investigated in our future work.

In Sec. 2, we define five audio types and four motion categories as our feature representation. Although other selection of audio/visual types (e.g. domain-specific features or types) can be similarly modeled by likelihood representation using Eq. (12), the representation that we adopted in this work is expected to depict common characteristics in a typical baseball game, regardless of where or when the game is held. For example, if we include one more audio type to model "loudness", this type would be mostly represented by audio energy function. Since automatic gain control (AGC) affects the final audio energy recorded by microphones and largely depends on different baseball fields, these environmental factors will make the modeling more difficult to be applied to a general baseball game and are very likely to bias final results as well.

Another issue should be mentioned is the unbalance subjective expression from audiences of a home-game and an away-game. The audience might cheer more loudly when the home-team scores, while booing against the guest-team's scoring. This subjectivity is still an open issue to be conquered. Nevertheless, we believe that incorporation of visual features could at least lessen the unbalance due to audience's subjectivity.

V. CONCLUSION

In this paper, we present an integrated framework for baseball game highlight extraction based on the proposed audio and motion feature representation methods. To better describe the content semantics, we propose using likelihood models to represent audio and motion features and combine these two information symmetrically. We build the highlight detection framework on HMM to capture the temporal transition of the combined audio-visual information, which leads to a promising result for extracting highlights from baseball games. With the proposed audio-visual likelihood models, we are able to express highlight contents in a more reliable and precise way. In addition, our experimental results demonstrate that the HMM-based framework with the integrated feature representation indeed extracts highlight with satisfactory performance without pre-defining any domain-specific scene types or highlight events.

Several issues will be further studied in our future work. We will investigate to eliminate the domination of static states in HMM. One of the possible ways is to constrain the state transition of HMM model. In addition, we will explore to incorporate the directional information into the motion representation. Since our proposed framework mainly relies on the statistical model, it is easy to extend the original framework with more representative features, such as object's moving trajectories. Moreover, as this work contains little domain-specific knowledge, it takes very little effort to apply the methodology to other applications or sport games.

REFERENCES

[1] D. Zhong and S.F. Chang, "Structure Analysis of Sports Video Using Domain Models," *Proc. ICME'01*, 2001.

[2] A. Ekin, A. Murat Tekalp and R. Mehrotra, "Automatic Soccer Video Analysis and Summarization," *IEEE Tran. Image Processing*, vol. 12, no. 7, Jul. 2003.

[3] V. Tovinkere and R. J. Qian, "Detecting Semantic Events in Soccer Games: Towards A Complete Solution," *Proc. ICME'01*, 2001.

[4] J. Assfalg, M. Bertini, A. Del Bimbo, W. Nunziati and P. Pala, "Detection and Recognition of Football Highlights using HMM," *Proc. ICECS'02*, 2002.

[5] G. Xu, Y.F. Ma, H.J. Zhang and S.Q. Yang, “A HMM Based Semantic Analysis Framework for Sports Game Event Detection,” *Proc. ICIP’03*, Sep. 2003.

[6] B. Li and I. Sezan, “Semantic Sports Video Analysis: Approaches and New Applications,” *Proc. ICIP’03*, Sep. 2003.

[7] T. Kawashima, K. Tateyama, T. Iijima and Y. Aoki, “Indexing of Baseball Telecast for Content-Based Video Retrieval,” *Proc. ICIP’98*, 1998.

[8] M. Petkovic, V. Mihajlovic and W. Jonker, “Techniques for Automatic Video Content Derivation,” *Proc. ICIP’03*, Sep. 2003.

[9] H. Pan, P. van Beek and M. I. Sezan, “Detection of Slow-Motion Replay Segments in Sports Video for Highlights Generation,” *Proc. ICASSP’01*, 2001.

[10] Y. Rui, A. Gupta and A. Acero, “Automatically Extracting Highlights for TV Baseball Programs,” *Eighth ACM International Conference on Multimedia*, pp. 105-115, 2000.

[11] Z. Xiong, R. Radhakrishnan, A. Divakaran and T. Huang, “Audio Events Detection Based Highlights Extraction from Baseball, Golf and Soccer Games in a Unified Framework,” *Proc. ICASSP 2003*, Apr. 2003.

[12] R. Leonardi, P. Migliorati and M. Prandini, “Semantic Indexing of Sports Program Sequences by Audio-Visual Analysis,” *Proc. ICIP 2003*, Sep. 2003.

[13] A. Divakaran, R. Radhakrishnan and K. A. Peker, “Motion Activity-based Extraction of Key-frames from Video Shots,” *Proc. ICIP’02*, Sep. 2002.

[14] X. Shao, C. Xu and M. S. Kankanhalli, “Automatically Generating Summaries for Musical Video,” *Proc. ICIP 2003*, Sep. 2003.

[15] Y. Gong and X. Liu, "Video Summarization and Retrieval using Singular Value Decomposition," *Multimedia Systems*, vol. 9, no. 2, Aug. 2003.

[16] M. Smith and T. Kanade, “Video Skimming and Characterization through the Combination of Image and Language Understanding Techniques,” *Proc. CVPR’97*, 1997.

[17] Y.F. Ma and H.J. Zhang, “A Model of Motion Attention for Video Skimming,” *Proc. ICIP 2002*, Sep. 2002.

[18] H. Sundaram, L. Xie and S.F Chang, “A Utility Framework for the Automatic Generation of Audio-Visual Skims,” *Proc. ACM Multimedia*, 2002.

[19] M.J. Roach, J.S.D. Mason and M. Pawlewski, “Video Genre Classification using Dynamics,” *Proc. ICASSP’01*, 2001.

[20] A. Bonzanini, R. Leonardi and P. Migliorati, “Event Recognition in Sport Programs using Low-Level Motion Indices,” *Proc. ICME’01*, 2001.

[21] N. Peyrard and P. Bouthemy, “Detection of Meaningful Events in Videos Based on A Supervised Classification Approach,” *Proc. ICIP’03*, Sep. 2003.

[22] Y.P. Tan, D. D. Saur, S. R. Kulkarni and P. J. Ramadge, “Rapid Estimation of Camera Motion from Compressed Video with Application to Video Annotation,” *IEEE Tran. Circuits and Systems for Video Tech.*, vol. 10, no. 1, Feb. 2000.

[23] P. Chang, M. Han and Y. Gong, “Extract Highlights from Baseball Game Video with Hidden Markov Models,” *Proc. ICIP 2002*, Sep. 2002.

[24] L. Xie, S.F. Chang, A. Divakaran and H. Sun, “Structure Analysis of Sports Video with Hidden Markov Models,” *Proc. ICASSP’02*, 2002.

[25] G. Xu, Y.F. Ma, H.J. Zhang and S.Q. Yang, "Motion-Based Event Recognition using HMM," *Proc. ICPR'02*, 2002.

[26] Y.L. Chang, W. Zeng, I. Kamel and R. Alonso, "Integrated Image and Speech Analysis for Content-Based Video Indexing," *Proc. ICMCS 1996*, 1996

[27] A. Albiol, L. Torres and J. Delp, "The Indexing of Persons in News Sequences using Audio-Visual Data," *Proc. ICASSP'03*, 2003.

[28] S.C. Chen, M.L. Shyu, W. Liao and C. Zhang, "Scene Change Detection by Audio and Video Clues," *Proc. ICME'02*, 2002.

[29] Z. Xiong, R. Radhakrishnan and A. Divakaran, "Generation of Sports Highlights using Motion Activity in Combination with a Common Audio Feature Extraction Framework," *Proc. ICIP 2003*, Sep. 2003.

[30] W. Hua, M. Han and Y. Gong, "Baseball Scene Classification using Multimedia Features," *Proc. ICME'02*, 2002.

[31] A. Hanjalic, "Generic Approach to Highlight Extraction from A Sport Video," *Proc. ICIP'03*, Sep. 2003.

[32] R. Dahyot, A. Kokaram, N. Rea and H. Denman, "Joint Audio Visual Retrieval for Tennis Broadcasts," *Proc. ICASSP'03*, 2003.

[33] Y. Gong, X. Liu and W. Hua, "Creating Motion Video Summaries with Partial Audio-Visual Alignment," *Proc. ICME'02*, 2002.

[34] C.C. Cheng and C.T. Hsu, "Content-Based Audio Classification with Generalized Ellipsoid Distance," *Proc. PCM 2002*, Dec. 2002.

[35] Y. Wang, Z. Liu, and J.C. Huang, "Multimedia Content Analysis," *IEEE Signal Processing Magazine*, pp. 12-36, Nov. 2000.

[36] A. M. Kondoz, *Digital Speech*, Wiley, 1994.

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

[38] J. R. Deller, J. H. L. Hansen and J. G. Proakis, *Discrete-Time Processing of Speech Signals*, IEEE Press, 2000.

[39] T. Hastie and R. Tibshirani, "Discriminant Analysis by Gaussian Mixtures," *J. Royal Statistical Society, series B*, 1996.

[40] T. Zhang and C.-C. J. Kuo, "Audio Content Analysis for On-line Audiovisual Data Segmantation and Classification," *IEEE Trans. Speech and Audio Processing*, vol. 9, no. 4, May 2001.

[41] M. A. T. Figueiredo and A. K. Jain, "Unsupervised Learning of Finite Mixture Models," *IEEE Tran. Pattern Analysis and Machine Intelligence,* vol. 24, no. 3, Mar. 2002.

[42] M. A. T. Figueiredo and A. K. Jain, "Unsupervised Selection and Estimation of Finite Mixture Models," *Proc. ICPR'00,* pp. 87-90, 2000.

[43] F. Dufaux and J. Konrad, "Efficient, Robust, and Fast Global Motion Estimation for Video Coding," *IEEE Tran. Image Processing,* vol. 9, no. 3, Mar. 2000.

[44] Y. T. Tse and R. Baker, "Global Zoom/Pan Estimation and Compensation for Video Compression," *Proc. ICASSP'91,* 1991.

[45] I. Koprinska and S. Carrato, "Temporal Video Segmentation: A Survey," *Signal Processing: Image Communication*, vol.16, p.477-500, 2001.

[46] A. Hanjalic, "Shot-Boundary Detection: Unraveled and Resolved?," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 12, no. 2, Feb. 2002.

[47] W.J. Heng and K.N. Ngan, "Shot Boundary Refinement for Long Transition in Digital Video Sequence," *IEEE Trans. Multimedia*, vol. 4, no. 4, Dec. 2002.

[48] L. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition," *Proc. Of the IEEE,* vol. 77, no. 2, Feb. 1999.

[49] D. Gatica-Perez, A. Loui and M.T. Sun, "Finding Structure in Home Videos by Probabilistic Hierarchical Clustering," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 13, no. 6, Jun. 2003.

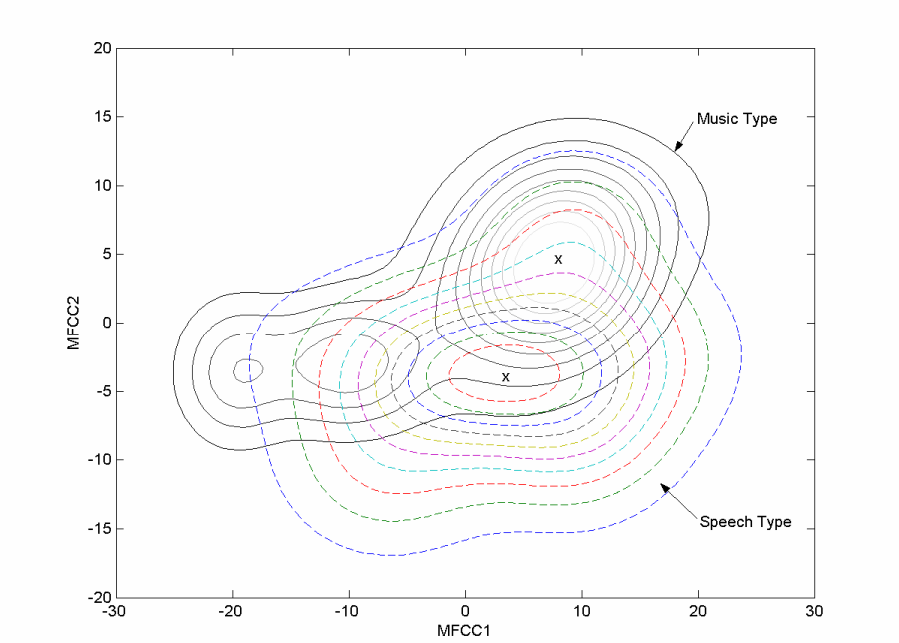


Figure 1. Illustration of the trained density functions for two audio types: music and speech. The solid lines denote the distribution of music, and the dotted lines denote the distribution of speech. The symbol "x" means the highest probability of the two distributions. Note that this figure is a projection from 14-dimension to a 2-dimensional feature space, and does not precisely reflect the actual overlap and distinction.

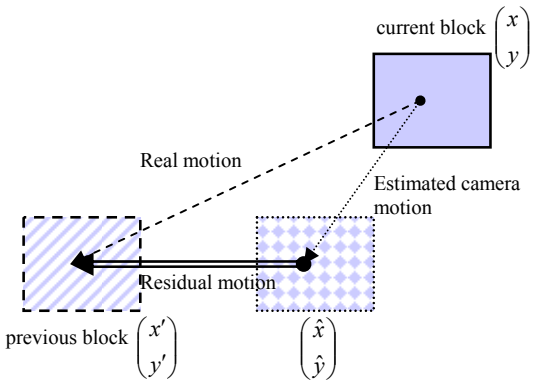
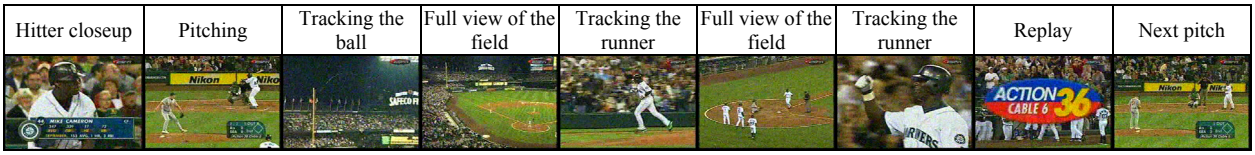
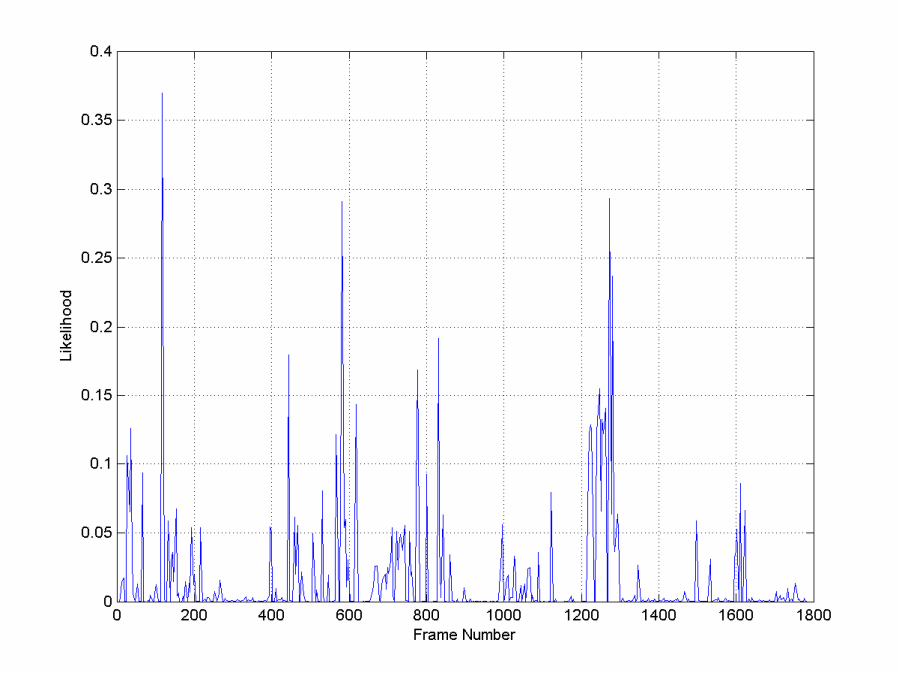


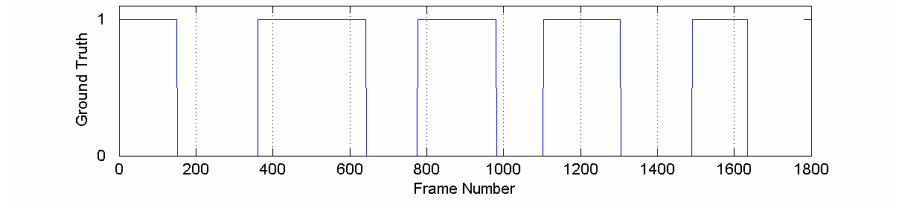
Figure 2. Estimation of object motion based on residual motion vectors.



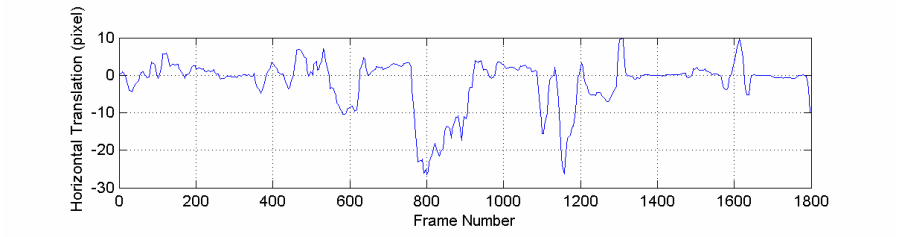
(a)



(b)



(c)



(d)

Figure 3. Illustration of the proposed motion likelihood model for camera pan. (a) The corresponding keyframes and descriptions for 8 distinguishable segments. (b) The resulting likelihood according to (12). (c) The manually labeled ground truth, where 1 and 0 indicate "pan" and "no pan" respectively. (d) The pan parameter estimated using (10).

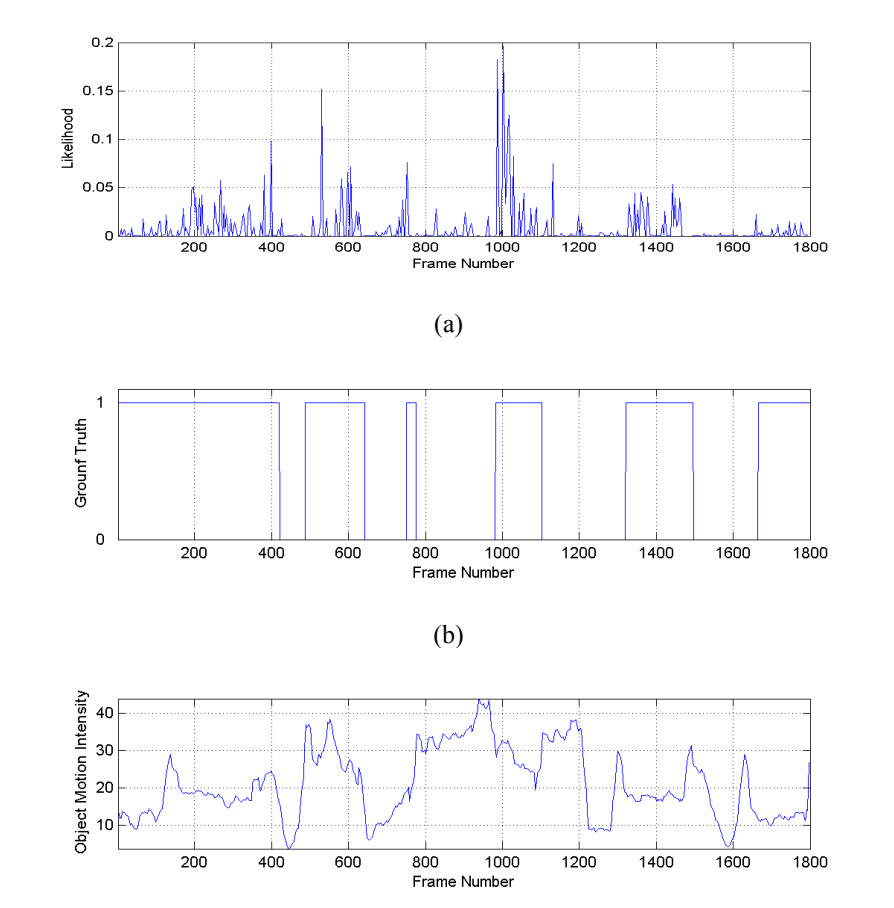


Figure 4. Illustration of the proposed likelihood model for object motion intensity. (a) The resulted likelihood curve using (22). (b) The ground truth with "1" indicates the existence of object motion. (c) The original estimated object motion intensity calculated by (21).

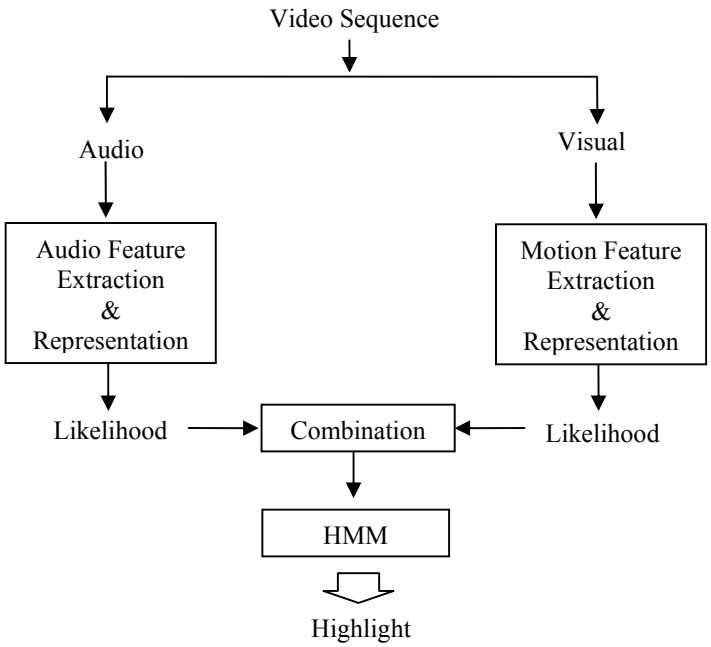


Figure 5. Flow chart of the proposed framework for highlight extraction.

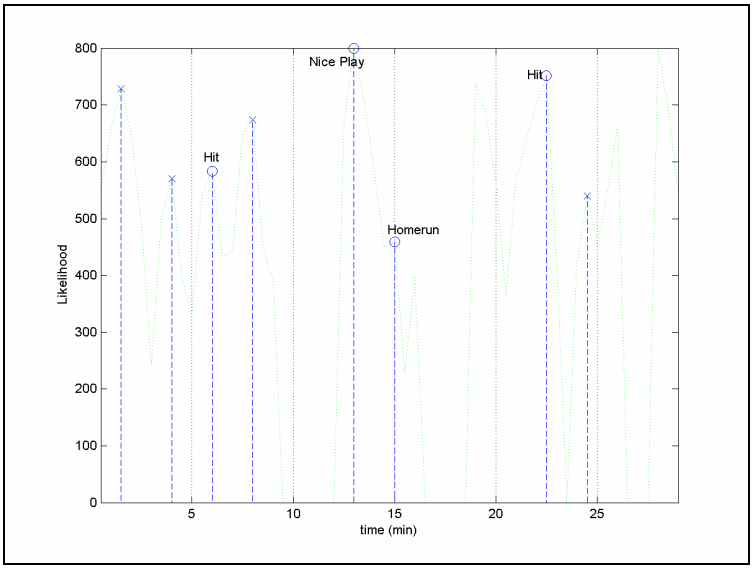


Figure 6. Highlight extraction results of test sequence T_0 based on the proposed integrated framework. The dotted line represents the resulting likelihood curve from our proposed HMM framework with integrated audio and motion features, and the dash line denotes the detected highlights, where correctly detected highlights are marked by circles and incorrectly detected ones are marked by cross.

Figure 7. The highlights extracted using the proposed integrated framework with test sequence T_0.

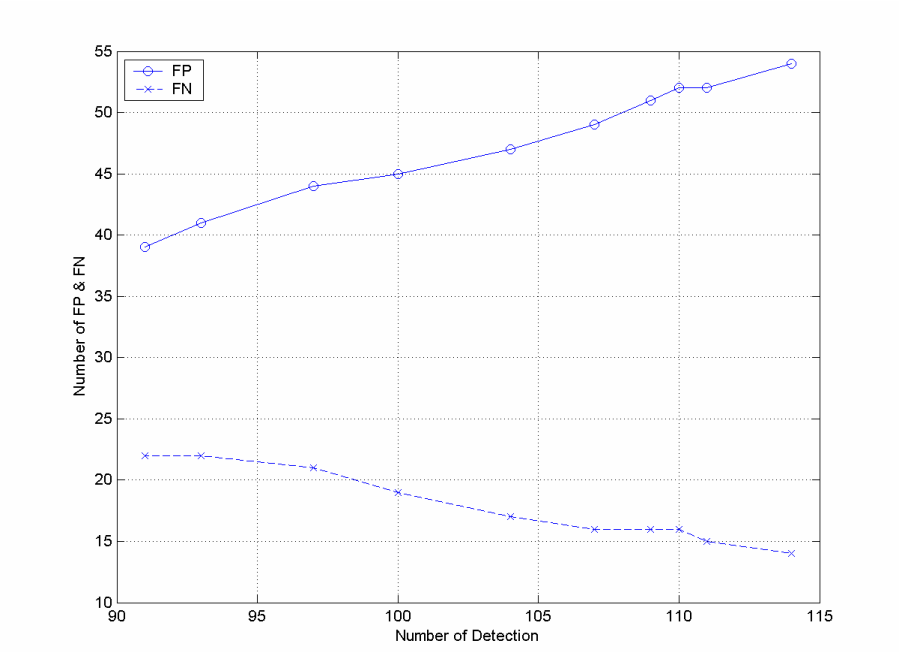


Figure 8. FP and FN versus number of detected highlights. The solid line denotes the number of FP under different thresholds whereas the dash line denotes the number of FN.

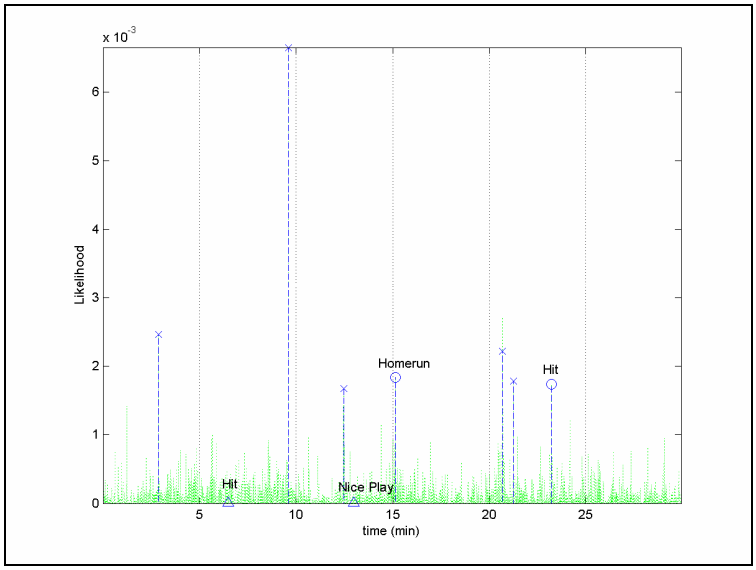


Figure 9. Highlight extraction results of test sequence T_0 based on deterministic detection approach. The dotted line represents the resulting likelihood curve from the deterministic framework (in (30)) with both audio and motion features, and the dash line denotes the detected highlights, where correctly detected highlights are marked by circles, incorrectly detected ones are marked by cross and the undetected highlights are marked by triangles.

Table 1. Highlight extraction results with test data set

Sequence	Ground Truth	Audio Feature Only			Visual Feature Only			Audio-Visual Combination		
		D	FP	FN	D	FP	FN	D	FP	FN
T_0	4	15	11	0	10	8	2	9	5	0
T_1	7	13	8	2	12	9	4	7	3	3
T_2	8	12	5	1	8	3	3	11	3	0
T_3	2	8	6	0	6	5	1	4	3	1
T_4	6	11	7	2	7	5	4	10	6	2
T_5	7	11	6	2	8	3	2	9	3	1
T_6	6	14	9	1	10	7	3	11	6	1
T_7	6	10	4	0	9	8	5	8	3	1
T_8	3	3	3	3	7	4	0	6	4	1
T_9	6	6	3	3	10	8	4	7	3	2
T_{10}	7	10	7	4	6	5	5	9	3	1
T_{11}	6	12	7	1	13	9	1	9	3	0
T_{12}	6	15	10	1	11	8	3	7	4	3
Total	**74**	**140**	**86**	**20**	**117**	**82**	**37**	**107**	**49**	**16**

Table 2. Comparison between deterministic reasoning and probabilistic inferring with test data

Sequence	Ground Truth	Deterministic			Probabilistic		
		D	FP	FN	D	FP	FN
T_0	4	6	4	2	9	5	0
T_1	7	27	20	0	7	3	3
T_2	8	38	33	3	11	3	0
T_3	2	40	38	0	4	3	1
T_4	6	6	1	1	10	6	2
T_5	7	6	3	4	9	3	1
T_6	6	9	7	4	11	6	1
T_7	6	22	17	1	8	3	1
T_8	3	26	23	0	6	4	1
T_9	6	31	27	2	7	3	2
T_{10}	7	29	24	2	9	3	1
T_{11}	6	9	3	0	9	3	0
T_{12}	6	8	4	2	7	4	3
Total	**74**	**257**	**204**	**21**	**107**	**49**	**16**

Table 3. Comparison between symmetric combination and visual-centric framework with test data

Sequence	Ground Truth	Visual Feature Only			Visual-Centric Method with Refinement by Audio			Audio-Visual Combination		
		D	FP	FN	D	FP	FN	D	FP	FN
T_0	4	10	8	2	8	6	2	9	5	0
T_1	7	12	9	4	12	9	4	7	3	3
T_2	8	8	3	3	8	3	3	11	3	0
T_3	2	6	5	1	6	5	1	4	3	1
T_4	6	7	5	4	5	3	4	10	6	2
T_5	7	8	3	2	5	1	3	9	3	1
T_6	6	10	7	3	8	6	4	11	6	1
T_7	6	9	8	5	8	7	5	8	3	1
T_8	3	7	4	0	7	4	0	6	4	1
T_9	6	10	8	4	10	8	4	7	3	2
T_{10}	7	6	5	5	6	5	5	9	3	1
T_{11}	6	13	9	1	11	7	2	9	3	0
T_{12}	6	11	8	3	11	8	3	7	4	3
Total	**74**	**117**	**82**	**37**	**105**	**72**	**40**	**107**	**49**	**16**