Real-Time Personalized Sports Video Filtering and Summarization

Di Zhong
Dept. of Electrical Engineering,
Columbia University
New York, NY 10027, US
dzhong@ee.columbia.edu

Raj Kumar
Dept. of Electrical Engineering,
Columbia University
New York, NY 10027, US
kumar@ee.columbia.edu

Shih-Fu Chang
Dept. of Electrical Engineering,
Columbia University
New York, NY 10027, US
sfchang@ee.columbia.edu

ABSTRACT
We demonstrate a real-time fully automated software system for filtering important events in sports video. Events represent occurrences of actions or state changes in video content. In the current prototype, we demonstrate detection of pitching in baseball and serving in tennis. For wireless video applications, we propose and apply a unique notion of content-based adaptive streaming, in which video encoding rate and media modality is dynamically varied according to the event filtering results. Our system includes an event detection module, an adaptive encoding module, and a buffer management module for adaptive streaming. We achieve the real-time performance by exploring compressed-domain techniques and multi-stage multi-resolution content-analysis processes.

Keywords
Video Filtering, Indexing, Summarization, Event Detection, MPEG-7

1. INTRODUCTION
Content filtering is important for reducing program duration or bandwidth of digital videos. Several new applications such as Personal Video Recorder and mobile video can be greatly enhanced by the content filtering capability. Real-time filtering over live content is particularly useful for time critical applications, such as sports.

There has been much work on video shot segmentation and object indexing. These approaches typically cannot satisfy the user needs because of the gap between the low-level analysis and high-level user needs.

Some works have been conducted to detect important events in videos of various domains, such as anchorperson reporting in news video and fast breaks basketball or scoring in soccer. However, the performance achieved by the ad-hoc methods can be improved; the real-time processing issue was not addressed; and/or the integration with wireless streaming was not addressed.

We present a real-time event detection and filtering system for sports video in this demo. We focus on the sports video initially due to the regular structure of the video data and the massive audience interest in practice.

Our system implements a unique concept, called *content-based filtering and adaptive streaming*, illustrated in Figure 1. Full-motion audio-video content is displayed during important periods (e.g., pitching and follow-up plays in baseball) while during non-important periods, only key frames, audio, and text are displayed. Such adaptation is particularly useful for mobile personalized applications. It allows for efficient usage of the bandwidth and can maximize the video quality of important video segments over bandwidth limited links. During the non-important segments, low-bit-rate data (e.g., audio and text) are transmitted and displayed. Users can still monitor the activities in the program by listening to the audio or viewing the key frames with text captions.

The core components in detecting sports events achieve very good performance (higher than 90% accuracy with real-time speed) in the current prototype. Detailed descriptions of the system architecture and component algorithms can be found in [1,2,4].

2. SYSTEM OVERVIEW
Figure 2 shows the system architecture of the content-based adaptive streaming system. The live video programs are received and then processed at the server or the filtering gateway (e.g., at mobile base station). The event detection and structure analysis module analyzes the input video and marks the important segments of video based on a preset criterion or customized user preference. One simple filtering criterion for baseball is to mark all pitching shots and subsequent shots showing continuing activities as important. For tennis, the criteria could be to select all serving shots and follow-up activity shots as important. For soccer, the important segments may include all the plays,

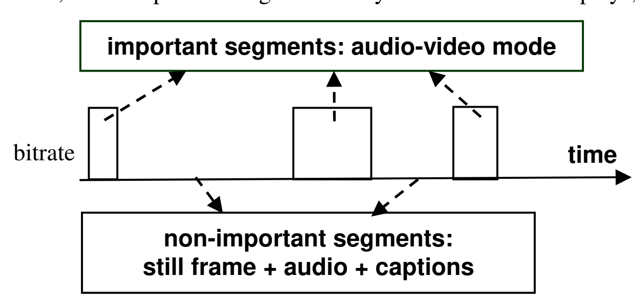

Figure 1. Content-based adaptive streaming of videos over bandwidth limited links.

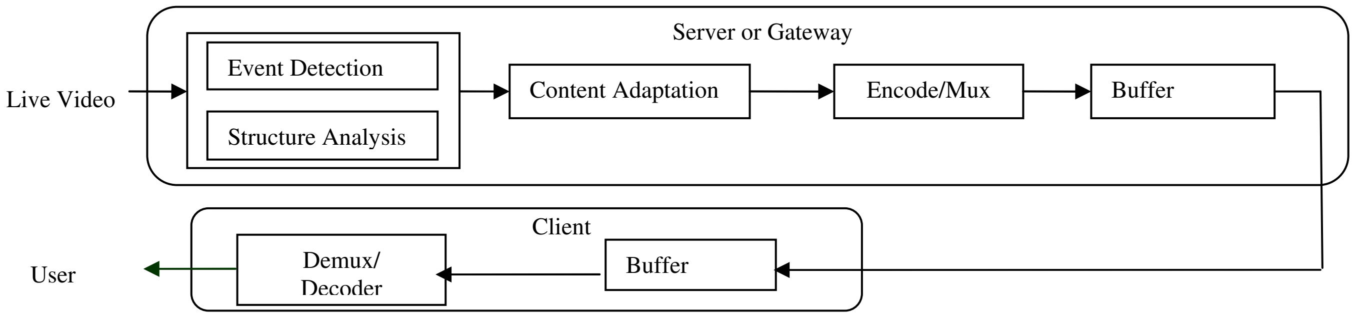

Figure 2. System Architecture of Content-Based Adaptive Video Filtering and Streaming

excluding the break segments between plays. Non-important shots may include commercials, commentators, and close-up shots of crowds, players, etc.

After the event detection module, the content adaptation module manipulates the actual audio-visual sequence according to the adaptation schedule. For example, if we adopt the adaptation schedule shown in Figure 1, video of all the important segments will be kept while the non-important segments will be replaced by key frames. The adapted content will be fed to the encoding module which is compliant with industry or standard formats. The actual encoding method may vary depending on the selected encoding and streaming format (e.g., MPEG-1, Real, or Microsoft). For some encoders, we can just encode the important segments plus the initial frames of the non-important segments. This is similar to the case of a live encoding session with the encoding operation turned on and off intermittently, according to the content importance.

We envision the client terminal to be a hand-held device or desk top PC with a low-bandwidth network link. To send the adaptive-rate video over a low-bandwidth link, buffer management modules are needed at the server and the client locations to smooth out the bursty rate. Finally, audio/video streams are demuxed, decoded, and displayed at the client according to the content adaptation schedule.

3. EVENT DETECTION

Given a specific domain such as baseball or tennis games, fundamental semantic units (FSU) are first identified by hand according to domain structure and semantic rules. FSU's represent an intuitive level of access and summarization of the video program. For example, in several types of sports (baseball, tennis, golf, basketball, soccer, etc), there is a fundamental level of content which corresponds to an intuitive cycle of activity in the game. For baseball, the FSU could be the period corresponding to a pitch and follow-up events (such as catch or run), a batter's play, or an inning. For tennis, the FSU could be the period corresponding to a serve and follow-up plays (such as stroke), or a set. For soccer, the FSU could correspond to a play. Organization of video data at these levels provide abstraction at multiple levels with flexible granularity.

Given the identified structure of videos in specific domains, we developed automatic tools to detect the boundaries of FSU's. We detect the FSU's by integrating multiple sources of information, such as

- recognition of the unique views associated with beginning of each FSU, and
- parsing of game status such as ball count and scores.

We demonstrate real-time detection of pitching and serving in baseball and tennis respectively. Figure 3 shows the architecture of our real-time serve view detection system. Details can be found in [1]. It includes multiple computing stages. First, the incoming video data is decomposed into shots by using a real-time compressed-domain shot detector implemented in software. In the second stage, a single key frame of each shot is extracted and frame-level features (such as color histogram) are extracted. Such frame-level features are matched against color models that have been constructed in a separate supervised learning process. The output of this stage includes candidate shots that match the unique visual views such as serving or pitching. Candidate shots passing the global visual model matching process are further examined by analyzing the constituent video objects and their spatio-temporal features (such as motion, geometry, and locations of the player and lines). Object-level constraints and rules applied in this stage can be specified by experts exploring the domain knowledge or learned in a semi-automatic way. An interactive learning system called Visual Apprentice [3] can be used. Such a system allows users to define structured scene models (e.g., a model for pitching scene with multi-level components corresponding to objects in the scene). It also learns features and classification constraints of component objects in the model through interactive labeling and training.

In some domains, the object-level features also allow for detection of some higher-level events. For example, as shown in [1], by tracking the location and moving pattern of the foreground moving object in tennis video, we were able to detect the number of strokes and the position of the player in relation to the net with reasonable accuracy. Such information is useful in the summary and navigation tools which will be described later.

Note automatic segmentation of video objects and analysis of object-level features involves complex computation. However, since such computation is used as a refined verification step and is not applied in every shot, real-time processing speed still can be achieved for the whole video.

The view detection process achieve very high accuracy (higher than 90%) in our current experiments using a few hours of tennis and baseball broadcast programs as test content and 10 minutes video each as training data.

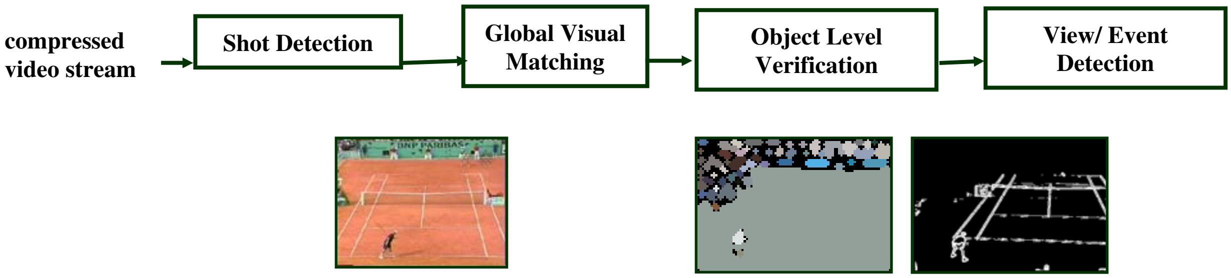

Figure 3. System Architecture for Automated Canonical View Detection in Tennis Video

4. PROTOTYPE STATUS

We are currently developing an integrated system combining the above event detection and adaptive streaming modules. Our demo testbed consists of a Pentium III class PC with two processors serving as the gateway, connected to a Compaq IPAQ with twisted-pair wiring serving as the physical layer and UDP/TCP/IP on top.

The gateway and client software is implemented in C++ using the Microsoft Format SDK. The gateway software consisted of a real-time structure parsing/event filtering module that identifies the FSUs and classifies the content importance. It also includes an encoding module to which the content filtering result is passed. The encoding module, which is implemented using the Microsoft Format SDK, uses the filtering information to compress the incoming stream into a variable rate MPEG-4 stream. The incoming stream is compressed into two target bitrates: a predetermined high target bitrate and a predetermined low target bitrate depending on whether the segment is classified as important or less important. The high target bitrate stream contains both video and audio, while the low target bitrate stream consists of a still-image and audio. The still-image is typically a frame extracted from the corresponding segment. This stream is passed to the PDA over UDP/IP and the twisted-pair link and is played back with a media-player again using Microsoft's Format SDK. The final playback stream uses the schedule similar to the one shown in Figure 1. For baseball, all the pitching segments plus their follow-up action shots are shown with the high bitrate, while the rest of the video are shown with the low bitrate. Similarly, for tennis, all the serving segments plus their follow-up action shots are shown at the high bitrate.

The testbed currently demonstrates the real-time performance of the software system in event filtering and adaptive streaming. The wireless link with bandwidth constraints has not been included and is currently simulated by the twisted-pair wiring. The incoming video sequences are pre-recorded sports programs and are read from the local disk in real-time during the test. Adding a hardware component to capture live videos broadcasted over the air and use them as the test input should not significantly affect the real-time performance of the system.

5. CONCLUSIONS

We present the content-based adaptive filtering and streaming system in this demo. The system detects and filters the important events in live broadcast video programs and adapts the incoming streams to variable rate according to the content importance. The importance can be defined based on program structure and user preference. The filtering results can be utilized to build a personalized summarization or navigation tool, in which users can navigate through the video program at multiple levels of abstraction efficiently. We achieve the real-time performance by combining our unique tools in compressed-domain processing, multi-feature fusing, and domain knowledge modeling. We also demonstrate high filtering accuracy in initial sports domains such as baseball and tennis.

Currently, we are investigating new implementations over actual wireless links and the feasibility of using new standard coding tools such as MPEG-4 scalable encoding. We are also developing techniques for detecting higher-level events (e.g., scoreing, new players) in the sports domain and application in other domains, such as soccer and film.

6. REFERENCES

[1] D. Zhong and S.-F. Chang, "Structure Analysis of Sports Video Using Domain Models, " IEEE Conference on Multimedia and Exhibition, Japan, Aug. 2001.

[2] P. Xu, S.-F. Chang, A. Divakaran, A. Vetro, and H. Sun, "Algorithms and System for High-Level Structure Analysis and Event Detection in Soccer Video," IEEE Conference on Multimedia and Exhibition, Japan, Aug. 2001.

[3] A. Jaimes and S.-F. Chang, "Model Based Image Classification for Content-Based Retrieval," SPIE Conference on Storage and Retrieval for Image and Video Database, Jan. 1999, San Jose, CA.

[4] S.-F. Chang, D. Zhong, and R. Kumar," Real-Time Content-Based Adaptive Streaming of Sports Video," Columbia University ADVENT Technical Report #121, July 2001. Also submitted to IEEE Workshop on Content-Based Access to Video/Image Library, Hawaii, Dec. 2001.