Structure Analysis of Sports Video Using Domain Models

Di Zhong and Shih-Fu Chang
{dzhong, sfchang}@ee.columbia.edu
Department of Electrical Engineering, Columbia University, NY, USA

Abstract

In this paper, we present an effective framework for scene detection and structure analysis for sports videos, using tennis and baseball as examples. Sports video can be characterized by its predictable temporal syntax, recurrent events with consistent features, and a fixed number of views. Our approach combines domain-specific knowledge, supervised machine learning techniques, and automatic feature analysis at multiple levels. Real time processing performance is achieved by utilizing compressed-domain processing techniques. High accuracy in view recognition is achieved by using compressed-domain global features as prefilters and object-level refined analysis in the latter verification stage. Applications include high-level structure browsing/navigation, highlight generation, and mobile media filtering.

1. Introduction

Video indexing and filtering is an important and challenging problem in view of the increasing amount of digital video content available. In recent years, visual features and objects have been studied for the indexing of generic video content [1]. While these features are useful in retrieving scenes based on certain visual similarity, they contain little information at the semantic level. To solve this problem, some works have explored the knowledge and constraints in specific domains and apply domain-specific rules as well as machine learning techniques. In [2] and [5], the story structure of a news program is re-constructed by detecting anchorperson views as well as commercials.

In this paper, we present a structure analysis framework for sports videos using domain models and supervised learning. Compared with other types of videos, sports videos have some unique characteristics. A sports game usually occurs in one specific playground, has a fixed number of camera views, contains abundant motion information and has well-defined content structures. Event detection in basketball and tennis videos has been studied in [4] and [5] respectively. But the performance remains to be improved and the real-time issue was not addressed.

Compared to existing work, our solution has the following unique features.

- An effective framework for view detection and structure parsing.
- Combination of domain-specific knowledge and generic machine learning techniques.
- A multi-stage scene detection framework combining frame-level view filtering and object-level analysis.
- Real time performance by using compressed-domain feature filtering.
- Higher accuracy demonstrated in specific sports domains such as tennis and baseball.

In Section 2, we will first discuss unique semantic content in sports videos. Our structural analysis system is described in Section 3, using tennis as an example. Experiment results in tennis and baseball videos are given in Section 4. Conclusion and future work are summarized in Section 5.

2. Content Structure in Sports Video

Sports video is a major part in most broadcasting TV programs, and has a large number of audience. Compared to other videos such as news and movies, sports videos have well-defined content structure and domain rules. A long sports game is often divided into a few segments. Each segment in turn contains some sub-segments. For example, in American football, a game contains two halves, and each half has two quarters. Within each quarter, there are many plays, and each play starts with the formation in which players line up on two sides of the ball. A tennis game is divided first into sets, then games and serves (as shown in Figure 1). In addition, in a sports video, there are a fixed number of cameras in the field that result in unique scenes during each segment. In tennis, when a serve starts, the scene is usually switched to the court view. In baseball, each pitch usually starts with a pitching view taken by the camera behind the pitcher. Furthermore, for TV broadcastings, there are commercials or other special information (e.g., score board, player's name) inserted between game sections. One objective of our work is automatic detection of fundamental views (e.g., serve and pitch) that indicate the boundaries of higher-level structures. Given the detection results, useful applications such as table of contents and structure summaries can be developed.

In this paper, we present a general framework and new algorithms to analyze the temporal structure of live broadcasted sports videos. We analyze the temporal structure by first defining and automatically detecting the re-current event boundaries, such as pitching and serving views. Such views are at a higher level than the traditional approaches using shots. They usually consist of unique visual cues, such as color, motion, and object layout.

Automatic detection of the fundamental views is

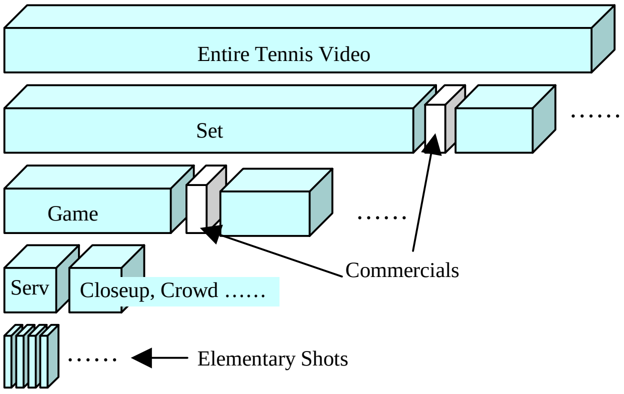


Figure 1. Temporal Structure of a typical tennis video program

achieved by using supervised learning and domain-specific rules. In the learning phase, filtering models based on compressed-domain features are learned. Such supervised learning process can be applied to any specific domain once the important views are defined. In the operation phase, a refined subset of filtering models are selected adaptively based on characteristics of the initial segment of the new test data. Such filtering models are used in matching the global features in each shot to detect candidate views. Candidate views are further verified using the object-level features that are derived using our automatic object segmentation tools [7]. Figure 2 shows the architecture of structure analysis system.

3. The Structure Analysis Framework

In this section, we describe our techniques for detecting basic units within a game, such as serves in tennis and pitching in baseball. These units usually start with a special scene. Simple color based approaches have been suggested in [4]. Based on our experiments, this type of approaches can only reach about 80 percent accuracy. Furthermore, as color information varies from game to game, adaptive methods need to be exploited. In order to achieve higher performance, we also added object-level verification to remove false alarms.

3.1 Color Based Adaptive Filtering

Color based filtering is applied to key frames of video shots. A key frame is the first I frame after each shot boundary. First, the filtering models are built through a clustering based training process. The training data should contain enough games so that a new game will be similar to some in the training set. Assume h_i, *i=1…,N* are color histograms of serve scenes in the training set. A k-means clustering is used to generate *K* models (i.e., clusters), $M_1,\ldots,M_K$, such that,

$$h_i \in M_j\ ,\quad if\ \ D(h_i, M_j) = \min_{k=1}^{K}(D(h_i, M_k)) \qquad (1)$$

where $D(h_i, M_k)$ is the distance between h_i and the mean vector of M_k, i.e. $H_k = \frac{1}{|M_k|}\sum_{h_i \in M_k} h_i$ and $|M_k|$ is the number of training scenes being classified into the model M_k. This means that for each model M_k, H_k is used as the representative feature vector. Note the color histogram was extracted from the DC thumbnail images directly from the I frame in the MPEG sequence. A compressed-domain shot segmentation was used to detect shot boundaries (abrupt and gradual shot changes) according to the changes and distribution of the DCT coefficients and motion vectors in the MPEG sequence.

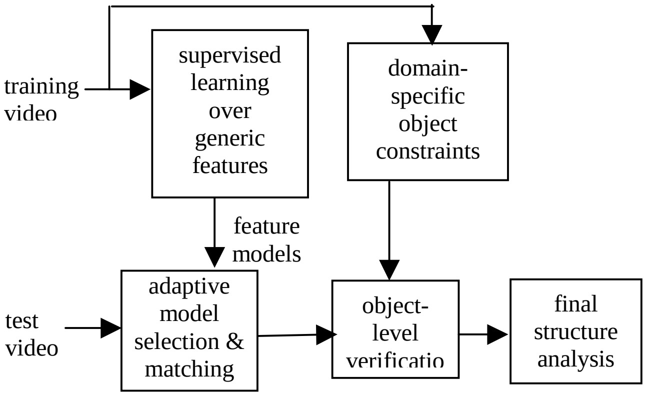

training video
supervised learning over generic features
domain-specific object constraints
feature models
test video
adaptive model selection & matching
object-level verificatio
final structure analysis

Figure 2. architecture for the view detection and structure analysis system

When a new game comes, a subset of proper models need to be chosen to detect serve scenes in the new data. This raises a typical egg-and-chicken problem, as we need to know serve scenes to select a correct model. To solve the problem, we detect the first *L* serve scenes using all models, $M_1,\ldots,M_K$. That is to say, all models are used in the filtering process. If one scene is close enough to any model, the scene will be passed through to subsequent verification processes (see Sections 3.2 and 3.3).

$$h_i' \in M_j\ ,$$

$$if\ \ D(h_i', M_j) = \min_{k=1}^{K}(D(h_i', M_k))\ \ and\ D(h_i', M_j) < TH \qquad (2)$$

where h_i' is the color histogram of the *i*th shot in the new video, and *TH* is a filtering threshold to accept shots with enough color similarity. Shot *i* is detected as a serve scene if the subsequent segmentation based verification, which will be described in Sections 3.2 and 3.3, is also successful, and we mark this serve as being founded by model M_j (i.e., classify the scene into the model M_j). If the verification fails, h_i' is removed from the set of M_j. Note the color-based models are simple. We use these simple models to detect candidate views and allow more false alarms. Removal of false alarms and accomplishment of high accuracy is done by the subsequent object-level verification.

After *L* serve scenes are detected, we find the model M_o, which leads to the search for the model with the most serve scenes.

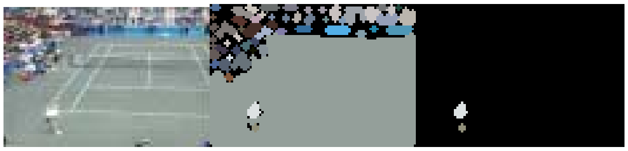
$$|M_o| = \max_{k=1}^{K}(|M_k|) \tag{3}$$

where $|M_k|$ is the number of incoming scenes being classified into the model M_k. In the filtering process for subsequent shots, only model M_o and a few of its closest neighboring models are kept and they are applied in the same way as that defined in Eq 2. The approach of choosing the domain model and its neighbors is based on the assumption that the play field in a game does not change significantly and so is its visual feature model. In the case when play field conditions may change notably (e.g., due to weather changes), the adaptive selection of the major model can be applied periodically during the processing of the whole program.

3.2 Object Segmentation Based Verification

Color histograms are global features that can be computed and compared faster than real time. However, with the color feature only, the detection accuracy is less than 80%. Many close-up scenes of playgrounds and replay scenes are likely to be detected as false positives. To improve detection accuracy, the salient region extraction and moving object detection methods we developed in [6] are utilized here to produce localized spatial-temporal features. Compared with global features, spatial-temporal features are more reliable and invariant for detecting given scene models. Especially in sports videos, special scenes are often made of several objects at consistent locations (e.g., players in the server scene in tennis).

In [6], we combined spatial consistency (color and edge) constraints to segment each frame into regions. Such regions are merged based on proximity and motion consistency. Merged regions are classified into foreground moving objects or background objects based on some rules of motions near region boundaries and long-term temporal consistency. Figure 2 shows a segmentation and moving object detection example. Figure 2(b) shows the segmented regions from an example frame. Note the court is segmented out as one large region, while the player closer to the camera is also extracted. The court lines are not preserved due to down-sampling of the frame size.



a. original b. regions c. moving object

Figure 2. An example of automatic region segmentation and moving object detection

Black areas shown in (b) are tiny regions being dropped. Figure 2(c) shows the moving object detection result after applying the motion object rules and checking the long-term temporary consistency. In this example, the result is very good, and only the approximate object corresponding to the player is detected. Sometimes a few background regions may also be detected as foreground moving objects. While this problem may be serious for applications requiring accurate player detection and tracking, here for the verification purpose as we will describe below, the important thing is not to miss the player object.

To achieve real-time performance, segmentation is performed on the down-sampled images of the key frame (which is chosen to be an I-frame) and its successive P-frame. The down-sampling rate used in our experiment is 4, both horizontally and vertically, which results in images with size 88x60. Motion fields are estimated using the hierarchical approach.

Following rules are applied in scene verification. First, there must be a large region (e.g. larger than two-thirds of the frame size) with consistent color (or intensity for simplicity). This large region corresponds to the tennis court. The uniformity of a region is measured by the intensity variance of all pixels within the region (Eq 4).

$$Var(p) = \frac{1}{N}\sum_{i=1}^{N}[I(p_i) - \bar{I}(p)]^2 \tag{4}$$

where N is the number of pixels within a region *p*. $I(p_i)$ is the intensity of pixel *I and* $\bar{I}(p)$ is the average intensity of region *p*. If *Var(p)* is less than a given threshold, the size of region *p* is examined to decide if it corresponds to the tennis court.

Secondly, the size and position of player are examined. The condition is satisfied if a moving object with proper size is detected within the lower half part of the previously detected large "court" region. In a downsized 88x60 image, the size of a player is usually between 50 to 200 pixels. As our detection method is applied at the beginning of a serve, and players are always at the bottom line to start a serve, the position of a detected player has to be within the lower half part of the court.

3.3 Edge Based Verification

One unique characteristic of serve scenes in tennis game is that there are horizontal and vertical court lines. Ideally if a camera shots straightforward from top-rear point of the court and all court lines are captured, rules for a complete court can be used to verify serve scenes with high precision. However, in a real scene, due to the camera panning and zooming, or object occlusion, usually not all court lines are viewable. Trying to match a full court will result in a low recall rate of serve scenes.

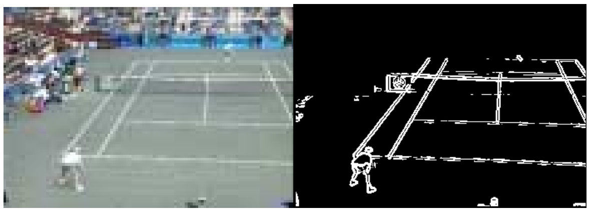


Figure 3. Edge detection within the court region

Since we already applied color based filtering and region based verification processes, relatively loosen constraints are enforced on court lines. An example of edge detection using the 5x5 Sobel operator is given in **Figure 3**. The edge detection is performed on a down-sampled (usually by 2) image and inside the detected court region only (see Figure 2b). Hough transforms are conducted in four local windows to detect straight lines. It greatly increases the accuracy in detecting straight lines by using local windows instead of a whole frame.

The verifying condition is that there are at least two vertical court lines and two horizontal court lines being detected. Note these lines have to be apart from each other for a certain distance, as noises and errors in edge detection and Hough transform may produce duplicated lines. This is based on the assumption that despite of camera panning, there is at least one side of the court, which has two vertical lines, being captured in the video. On the other hand, camera zooming will always keep two of three horizontal lines, i.e., the bottom line, middle court line and net line, in the view.

4. Experiment Results

We applied the above view detection system to tennis and baseball videos, with different color models and verification rules respectively. Our experiment tests on a one-hour tennis video and a one-hour baseball video. Training data includes several short segments (a few minutes to 10 minutes) from different broadcast games and channels. Table 1 shows the results when initial segment of the test data is included in the training set. Table 2 shows the results when the initial segment of the test program is excluded from the training data. The results indicate that the performance is very good (with both precision and recall higher than 90%) and can handle new content from different broadcast programs in our test set.

Table 1. Detection results of serve and pitch scenes (initial segment of the test video included in the training set)

	Ground truth	# of Miss	# of False
Tennis (serve)	89	7	2
Baseball (pitch)	93	3	4

Table 2. Detection results of serve and pitch scenes (initial segment of the test video excluded from the training set)

	Ground truth	# of Miss	# of False
Tennis (serve)	74	6	1
Baseball (pitch)	57	1	5

These results are very good compared to approaches using colors only. Based on our experiments, previously proposed approaches using color histogram filtering can only achieve about 80% precision rate (to obtain near 100% recall). Furthermore, despite of using advanced segmentation and feature extraction, our scene detection and verification process is performed in real time when tested over CIF-sized 30 fps MPEG-1 video on a regular single CPU PC. We conjecture that real-time performance is feasible for video with higher resolution if the software implementation is further optimized.

5. Conclusions

In this paper, we presented an effective framework for structure analyzing and scene detection for sports videos, using tennis and baseball as examples. It combines domain-specific knowledge, supervised learning techniques, and automatically segmented objects and features from both compressed and uncompressed domains. Real time processing performance is achieved by doing the intensive matching using the compressed-domain features, while high accuracy is achieved by using object-level detailed analysis. The framework can be flexibly applied to different domains by following the same approach and framework. Applications include high-level structure summarization/navigation and highlight filtering of long video programs.

References

1. S.-F. Chang, Q. Huang, T. Huang, A. Puri, and B. Shahraray, "Multimedia Search and retrieval," book chapter in Advances in Multimedia: Systems, Standards, and Networks, New York: Marcel Dekker, 1999.
2. Q. Huang, Z. Liu, A. Rosenberg, “Automated Semantic Structure Reconstruction and Representation Generation for Broadcast News”, IS&T/SPIE Conference on Storage and Retrieval for Image and Video Database VII, San Jose, California, Jan 1999.
3. D. D. Saur, T.-P. Tan *et al*. “Automated Analysis and Annotation of basketball Video”, Proceedings of SPIE's Electronic Imaging conference on Storage and Retrieval for Image and Video Databases V, Feb 1997.
4. G. Sudhir, J. C.M. Lee and A.K. Jain, “Automatic Classification of Tennis Video for High-level Content-based Retrieval”, Proc. Of the 1998 International Workshop on Content-based Access of Image and Video Database, January 3, 1998 Bombay, India.
5. H. J. Zhang *et al*, “Automatic Parsing and Indexing of News Video”, Multimedia Systems, 2 (6), pp. 256-266, 1995.
6. D. Zhong and S.-F. Chang, "Long-Term Moving Object Segmentation and Tracking Using Spatio-Temporal Consistency," IEEE International Conference on Image Processing, Greece, Oct. 2001.