SHOT TYPE CLASSIFICATION BY DOMINANT COLOR FOR SPORTS VIDEO SEGMENTATION AND SUMMARIZATION

Ahmet Ekin[1] *and A. Murat Tekalp*[1,2]

[1]Department of Electrical and Computer Engineering, University of Rochester, Rochester, NY
[2] College of Engineering, Koc University, Istanbul, Turkey
{ekin,tekalp}@ece.rochester.edu
http://www.ece.rochester.edu/{~ekin,~tekalp}

ABSTRACT

This paper introduces a novel generic framework for sports video processing by using the common feature of most sports: the dominant color of the field. This dominant field color is automatically detected and updated to compensate for the lighting and weather changes by a robust dominant color region detection algorithm. We also introduce new shot type classification algorithms for soccer and basketball. Finally, we use shot-based low-level features for domain-specific high-level applications. Specifically, the system detects soccer goals and summarizes soccer games in *real-time,* and it enables basketball fans to skip fouls, free throws, and time-out events by segmenting a basketball game into plays and breaks.

1. INTRODUCTION

Sports video distribution over various networks in *real* or *near real-time* should contribute to quick adoption and widespread usage of multimedia services worldwide, because sports video appeals to large audiences. Processing of sports video, for example detection of important events and creation of summaries, makes it possible to deliver the content also over narrowband networks, such as Internet and wireless, since the valuable semantics generally occupy only a small portion of the whole content. Therefore, sports video processing must yield *semantic* results in *real,* or *near real-time,* and should be *automatic,* due to, otherwise, the intimidating size of the whole content.

In this paper, we generalize the framework introduced in [1] by processing both soccer and basketball video. Specifically, this paper introduces: 1) Generic dominant color region detection algorithm, which is robust in both outdoor and indoor scenes, for field region detection, 2) a new shot type classification algorithm for basketball that *automatically* detects all necessary threshold values, 3) adaptive play-break detection algorithm for basketball video, where we use the semantics associated with shot length. The use of the semantics related to shot length is, to the authors' knowledge, is the first in the sports video domain.

This work has been supported in part by the National Science Foundation under grant number IIS-9820721 and Eastman Kodak Company.

The paper follows the flowchart of the proposed framework in Fig. 1. However, due to space limitation, we refer the reader to [1] for shot boundary detection algorithm and [2] for slow-motion detection algorithms. In the next section, we explain the automatic detection of the dominant color region. Then, in Sec. 3, we describe domain-specific shot type classification algorithms. Sec. 4 introduces the algorithms for high-level processing of sports video, and it is followed by results and conclusion.

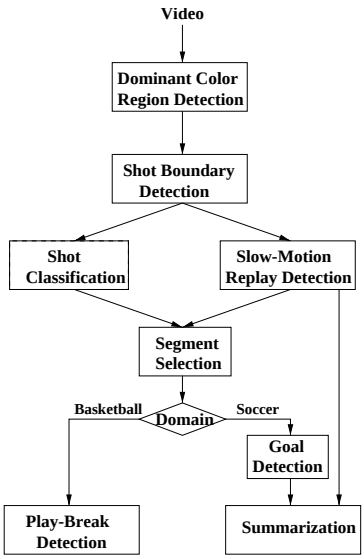

Figure 1: The flowchart of the proposed system

2. DOMINANT COLOR REGION DETECTION

In most sports, the field is characterized by *one distinct dominant color*. The statistics of this dominant color, in the selected color space, are learned by the proposed system at start-up, and then, if necessary,[1] automatically updated to adapt to temporal variations that may be caused by weather and/or lighting changes.

[1]Please see [1] for the algorithm for automatic update of dominant color statistics.

The dominant field color is described by the mean value of each color component, which are computed around their respective histogram peaks. The system builds the one-dimensional color histograms from the first one or more frames, and finds the peak index, i_{peak}, for each histogram. Then, an interval about each histogram peak is defined, where the interval boundaries (i_{min}, i_{max}) correspond to the closest indices to the peak index that have less pixel count than the threshold value, which is computed as %20 of the histogram peak. Finally, the mean color in the detected interval is computed for each color component.

Field colored pixels in each frame are detected by finding the distance of each pixel to the mean color. We use a *robust* cylindrical metric [3] for HSI (hue - saturation - intensity) space. In HSI space, achromaticity must be handled with care. If the estimated saturation and intensity means fall in the achromatic region, only intensity distance in Eq. 1 is computed for *achromatic* pixels. Otherwise, both Eq. 1 and Eq. 2 are employed for *chromatic* pixels in each frame.

$$d_I(j) = |I_j - I_{mean}| \quad (1)$$

$$d_{chroma}(j) = \sqrt{(S_j)^2 + (S_{mean})^2 - 2S_j S_{mean} cos(\theta)} \quad (2)$$

$$d_{cylindrical}(j) = \sqrt{(d_I)^2 + (d_{chroma})^2} \quad (3)$$

In the equations, S, and I refer to saturation and intensity, respectively, j is the j^{th} pixel, and θ is the minimum absolute difference between the hue values, i.e. θ is limited to $[0, \pi]$. The field region is defined as those pixels having $d_{cylindrical} < T_{color}$ where T_{color} is determined automatically from the first few frames given the rough percentage of field colored pixel ratio.

3. SHOT TYPE CLASSIFICATION

In cinematography, shots are generally classified into three: long, medium, and close shots [4]. In a specific domain, shot type information, when combined with other features, conveys interesting semantics. Motivated by this observation, we classify sports video shots, immediately after the detection of a shot boundary by the algorithm in [1], into three classes: 1) Long shots, 2) In-field medium shots, and 3) Close-up or out-of-field shots. The definitions and characteristics of each class are given below:

- **Long shot:** A long shot displays the global view of the field as shown in Fig 2 (a) and (b); hence, a long shot serves for accurate localization of the events on the field.
- **In-field medium shot:** A medium shot, where a whole human body is usually visible, is a zoomed-in view of a specific part of the field as in Fig 2 (c) and (d).
- **Close-up or Out-of-field Shot:** A close-up shot usually shows above-waist view of a player (Fig 2 (e)). The audience, coach, and other shots are denoted as out-of-field shots (Fig 2 (f)). We analyze both out-of-field and close-up shots in the same category due to their similar semantic meaning.

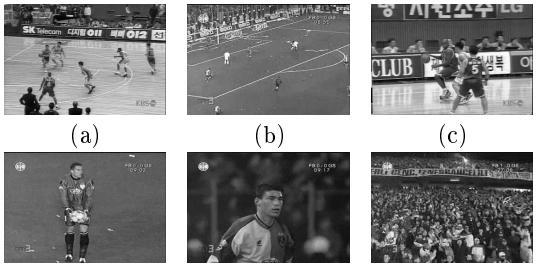

Figure 2: Shot types: (a,b) Long, (c,d) in-field medium, (e) close-up, and (f) out-of-field shots

Classification of a shot into one of the above three classes is based on spatial features. We find the shot type of every frame in a shot and assign the shot class to the label of the majority of frames. In order to find the frame view, *dominant colored pixel ratio in a frame, G*, is computed. In [5], an intuitive approach is used, where a low G value ($G < T_{closeUp}$) in a frame corresponds to a close-up or out-of-field view, while high G value ($G > T_{medium}$) indicates that the frame is a long view, and in between, a medium view is selected. Although the accuracy of this approach is sufficient for some applications, such as play-break detection in basketball, a more accurate method is needed for higher-level applications. In the following, we introduce a more robust shot type classification algorithm for soccer. Then, we explain the problems and approximate solutions for basketball shot classification.

3.1. Shot Type Classification for Soccer

In soccer, by using only grass colored pixel ratio, medium shots with high G value (when $G > T_{medium}$) will be mislabeled as long shots. The error rate due to this approach depends on the broadcasting style and it usually reaches intolerable levels for the employment of higher level algorithms. Therefore, we propose a compute-easy, yet very efficient, cinematographic measure for the frames with a high G value. We define regions by using *Golden Section* spatial composition rule [6], which suggests dividing up the screen in 3:5:3 proportion in both directions, and positioning the main subjects on the intersection of these lines. We have revised this rule for soccer video, and divide the *grass region box* instead of the whole frame. *Grass region box* can be defined as the minimum bounding rectangle (MBR), or a scaled version of it, for grass colored pixels. In Fig. 3, the examples of the regions obtained by *Golden Section* rule is displayed on several medium and long views. In the regions R_1, R_2, and R_3 in Fig. 3 (d) and (f), we have found the two features below, which measure the distribution of the grass colored pixels in medium and long views, the most distinguishing: G_{R_2}, the grass colored pixel ratio in the second region, and R_{diff}, the average of the sum of the absolute grass color pixel differences between R_1 and R_2, and between R_2 and R_3, that is $R_{diff} = \frac{1}{2}\{|G_{R_1} - G_{R_2}| + |G_{R_2} - G_{R_3}|\}$.

We employ a Bayesian classifier using the above two features to determine the correct class label. As mentioned before, the algorithm also employs two thresholds, $T_{closeUp}$

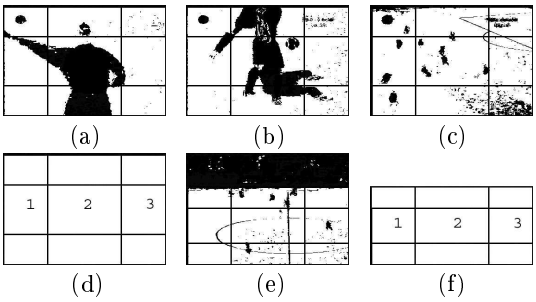

Figure 3: Examples of Golden Section spatial composition in (a-b) medium and (c-e) long views, the resulting grass region boxes and the regions are shown in (d) and (f) for (a-c) and (e), respectively

and T_{medium} to determine the frame view label. These two thresholds are roughly initialized to 0.1 and 0.4 at the start of the system, and as the system collects more data, they are updated to the minimum of the grass colored pixel ratio, G, histogram as suggested in [5].

3.2. Shot Type Classification for Basketball

The physical structure of a basketball field results in: 1) visible out-of-field region in long shots, i.e. a basketball long shot usually includes some part of the stadium and 2) the possibility of the painted regions under each basket and in the center field. Therefore, *Golden Section* rule, or a similar rule, does not distinguish medium shots from long shots. However, the final application in basketball is play-break detection; hence, we favor long shots over medium shots in order not to miss any play shot, i.e., we try to achieve high recall rate in long shot detection.

We quantize the dominant colored pixel ratio in a frame, G, to find the class of each shot by the thresholds, ($T_{closeUp}$, T_{medium}, $T_{highLong}$). The first two of these thresholds have the same meaning as in soccer, and the last one adds an upper bound to long shot dominant colored pixel ratio, since in basketball video, long shots usually do not have excessive amount of field colored pixels due to the first observation above. That is, a long shot is defined as $T_{medium} < G <= T_{highLong}$ and a medium shot is defined as $T_{closeUp} < G <= T_{medium}$ or $G > T_{highLong}$. The thresholds are automatically learned as explained in the following: Let us assume that the system does not have any knowledge about the camera locations and the field, i.e. painted/unpainted. Starting with the roughly initialized thresholds, such as (0.1, 0.25, 0.75), as the system detects shot boundaries,[2] shot length is used as a feature to localize long play shots. Each shot with a long length and high dominant colored pixel ratio (since at this time we do not know if T_{medium} is an accurate initialization, we compare the average G across the whole shot against $T_{closeUp}$, $G_{avg} > T_{closeUp}$) is labeled as a long shot. The dominant colored pixel ratio, G, histogram, H_{Long}, of the long shots is constructed from those shots that pass the above shot length and G_{avg} conditions. Similarly, we can construct the histograms for medium shots, H_{medium} by assuming the short length shots with high G_{avg} to be medium shots, and those shots with low G_{avg} without any condition on shot length to be close-up or out-of-field shots and construct $H_{closeUp}$. Once these histograms are obtained, the thresholds are determined (and updated at will as the game progresses) to minimize the error due to the misclassification. As already noted, we have a high recall requirement over long shots; therefore error due to the mislabeling of medium shots as long shots is tolerated while the opposite is highly penalized.

[2]Shot boundaries can be detected by either our proposed algorithm for sports video [1] or any generic robust algorithm, such as CueVideo [7] of IBM, at the expense of lower accuracy. The proposed algorithm for basketball is not sensitive to the latter case.

4. HIGH-LEVEL PROCESSING

4.1. Soccer Goal Detection and Summarization

The occurrence of a goal is generally followed by a special pattern of cinematic features. A goal event leads to a break in the game. During this break, the producers convey the emotions on the field to the TV audience and show one or more replay(s) for a better visual experience. The emotions are captured by one or more close-up views of the actors of the goal event, such as the scorer and the goalie, and by shots of the audience celebrating the goal. For a better visual experience, several slow-motion replays of the goal event from different camera positions are shown. Then, the restart of the game is usually captured by a long shot. Between the long shot resulting in the goal event and the long shot that shows the restart of the game, we define *a cinematic template* that should satisfy the following requirements:

- *Duration of the break:* A break due to a goal lasts no less than 30 and no more than 120 seconds.
- *The occurrence of* **at least one** *close-up/out-of-field shot:* This shot may either be a close-up of a player or out-of-field view of the audience.
- *The existence of* **at least one** *slow-motion replay shot:* The goal play is always replayed one or more times.
- *The relative position of the replay shot:* The replay shot(s) follow the close-up/out-of-field shot(s).

The search for goal event templates starts by detecting the slow-motion replay shots. For every slow-motion replay shot, the algorithm finds the long shots that define the start and the end of the corresponding break. These long shots must indicate a play that is determined by a simple duration constraint, i.e. long shots of short duration are discarded as breaks. Finally, the conditions of the template are verified to detect goals. The proposed "cinematic template" models goal events very well, and the detection runs in real-time (on a Pentium 4, 1.6GHz machine) with a very high recall rate. As a result, the proposed framework is able to generate two types of *real-time* soccer summaries: 1) All slow-motion replay shots in a game and 2) all goals in the game.

4.2. Basketball Play/Break Detection

A basketball game coverage usually lasts for around 2 hours although the game itself is less than an hour. Therefore, the

segmentation of a game into plays and breaks may reduce the coverage time by 50% without any loss in the content. Furthermore, play and break detection in basketball enables a hierarchical semantic analysis of the domain.

We propose to use shot length and shot type features to determine play and break segments. As explained in Sec. 3.2, a long shot with long length indicates a play. In basketball, a team is allowed 24 sec play time, but most plays last shorter than 24 sec and longer than 10 sec; hence, we found out that the shot length for a play to be 7 sec to capture all long play shots after a statistical analysis over the detected shots of a 10-minute-game. A play is captured as a long shot, but during a play, or immediately after an interesting action on the field, medium shots or player close-ups may be shown, and the omission of those shots as breaks results in abrupt and visually unpleasant transitions between the generated play segments. Therefore, if the interval between two play shots lasts shorter than T_B sec, it should be labeled as play. We believe the users should be able to change T_B, the allowable break time between plays. The value of T_B determines the actual compression rate; as T_B increases, the length of the play segment increases.

5. RESULTS

5.1. Soccer

17 sequences, more than 13 hours of soccer video, are used for the experiment. Each sequence, in full length, is processed to locate shot boundaries, shot types, and replays. The proposed algorithm detected 27 goals out of 30 in the dataset; thus, on the average, it achieved a 90.0% recall and 45.8% precision rates. We believe that the three misses are more important than false positives, since the user can always fast-forward false positives, which also do have semantic importance due to the replays. Two of the misses are due to the inaccuracies in the extracted shot-based features, and another one, where the replay shot is broadcast minutes after the goal, is due to the deviation from the goal model. The false alarm rate is directly related to the frequency of the breaks in the game. Breaks due to fouls, throw-ins, offsides, etc. with one or more slow-motion shots may generate cinematic templates similar to that of a goal. The inaccuracies in shot boundaries, shot types, and replay labels also contribute to the false alarm rate.

The compression rate for the summaries varies with the requested format. On the average, 12.78% of a game is included to the summaries of all slow-motion segments, while the summaries consisting of all goals, including all false positives, only account for 4.68%, of a complete soccer game. These rates correspond to the summaries that are less than 12 and 5 minutes, respectively, of an approximately 90-minute game.

5.2. Basketball

Table 1 shows detection (D) and false alarm rate (FA) in basketball sequences for a fixed T_B. In the experiment, ground truth play-break segments achieve 30.9, 22.8, and 18.9% compression rates for *SpainB, NCAAB,* and *KoreaB* sequences, respectively. Although Table 1 shows that the optimal T_B value is dependent on a particular sequence, i.e. game, and a particular broadcaster, it is still possible to conclude that play event summaries generated for T_B values at the lower half of 10-20 sec. interval consist of 92-95% of all play events with a very low false alarm rate, while the upper half of the same interval results in summaries of almost all, 98-100%, play events with some false positives.

T_B (sec)	SpainB D, FA	NCAAB D, FA	KoreaB D, FA
5	0.92, 0.0	0.92, 0.08	0.81, 0.0
10	0.92, 0.0	0.95, 0.08	0.92, 0.0
20	0.98, 0.23	1.0, 0.39	1.0, 0.18
30	1.0, 0.25	1.0, 0.39	1.0, 0.18

Table 1: Performance of the proposed play-break event detection algorithm for basketball video (D: play event detection rate, FA: play event false alarm rate)

6. CONCLUSION

We presented a generic framework that utilizes the dominant color of the field to detect shot classes in a variety of sports, such as soccer and basketball. Then, domain-specific high-level algorithms were introduced for soccer goal detection and summarization, and basketball play-break detection. The former enables the users to stream live games over narrowband channels, while the latter lets them skip break segments, due to time-outs, fouls, and free-throws.

REFERENCES

[1] A. Ekin and A.M. Tekalp, "Automatic soccer video analysis and summarization," *IS&T SPIE Sto. and Retr. for Med. Dat.,* 2003 (also at www.ece.rochester.edu/~ekin/publications.html).

[2] H. Pan, P. van Beek, and M.I. Sezan, "Detection of slow-motion replay segments in sports video for highlights generation," *ICASSP,* 2001.

[3] K. N. Plataniotis and A. N. Venetsanopoulos, *Color image processing and applications,* Springer-Verlag, Berlin, Germany, pp. 260-275, 2000.

[4] F. Shook, *Television field production and reporting,* 3^{rd} Ed., Allyn&Bacon Pub., 2000.

[5] P. Xu et. al., "Algorithms and system for segmentation and structure analysis in soccer video," *ICME,* Aug. 2001.

[6] G. Millerson, *The technique of television production,* 12^{th} Ed., Focal Publishers, 1990.

[7] A. Amir et al., "Using audio time scale modification for video browsing," HICSS-33, 2000.