

VIDEO CONTENT REPRESENTATION FOR SHOT RETRIEVAL AND SCENE EXTRACTION

TONG LIN⁺

National Laboratory of Machine Perception
Peking University
Beijing 100871, China
Email: tonylin0602@sina.com

HONG-JIANG ZHANG

Microsoft Research, China
49 Zhichun Road
Beijing 100080, China
Email: hjzhang@microsoft.com

QING-YUN SHI

National Laboratory of Machine Perception
Peking University
Beijing 100871, China
Email: tonylin0602@sina.com

In this paper, we present a novel scheme on video content representation by exploring the spatio-temporal information. A pseudo-object-based shot representation containing more semantics is proposed to measure shot similarity and an force competition approach is proposed to group shots into scene based on content coherences between shots. Two content descriptors, color objects: Dominant Color Histograms (DCH) and Spatial Structure Histograms (SSH), are introduced. To represent temporal content variations, a shot can be segmented into several subshots that are of coherent content, and shot similarity measure is formulated as subshot similarity measure that serves to shot retrieval. With this shot representation, scene structure can be extracted by analyzing the splitting and merging force competitions at each shot boundary. Experimental results on real-world sports video prove that our proposed approach for video shot retrievals achieve the best performance on the average recall (AR) and average normalized modified retrieval rank (ANMRR), and Experiment on MPEG-7 test videos achieves promising results by the proposed scene extraction algorithm.

1. Introduction

Video is a temporal media with huge amount of data, which cannot be easily organized and managed. Automatic video partitioning is the solution to segment video data into hierarchical structures, i.e. shots and scenes. A shot is a sequence of frames that are recorded contiguously, usually ended with a camera cut or an edit special effect. A video scene is referred to a group of consecutive shots taken place in the same location, or more generally, they share the same semantics in terms of time, place, objects or events. Video structure hierarchy is shown in Fig 1.

⁺ The work reported in this paper was mainly performed when this author was working at Microsoft Research, China as a research intern.

To date, most works on content-based video retrieval (CBVR)¹ deal with problems like shot boundary detection and keyframe extraction. The compact representation of video shot content for shot similarity measure remains one of the most challenging issues, which can be further used for shot-based video applications, like shot retrieval and shot grouping. In the current literatures^{1,2,3,4,5,6,7}, video shots are mostly represented by keyframes. Low-level features such as color, texture, and shape are extracted directly from keyframes for indexing and retrieval. For efficiency reason, video retrieval is usually tackled in a similar way as image retrieval. Such strategy, however, is ineffective since spatio-temporal information existing in videos is not fully exploited.

Recently, the temporal relationships among keyframes are explored to represent shot content. Zhao proposed a breakpoint detection algorithm⁸ to compactly select keyframes, and the nearest feature line (NFL) approach⁹, rather than comparing corresponding keyframes, is exploited to compute shot similarity. In¹⁰, Ngo proposed a motion-driven keyframe selection and computing by analyzing the temporal slice patterns. A static shot can be represented by any frame within it; a zoom shot can be represented by the first and last frame; a mosaic image can be constructed as a new representative frame for a pan or tilt shot; targeted objects can be extracted for a object tracking shot. Although those works have achieved reasonably good results, there has no work on how to utilizing the video objects (or pseudo-objects) to exploit the spatio-temporal events for video content representation and retrieval.

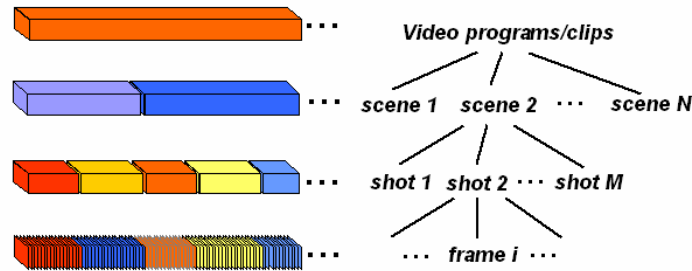


Fig 1: the video-scene-shot-frame hierarchy.

A scene is defined as one or more consecutive shots that they are semantically correlated⁷, or they all share the same “content” in terms of action, place and time³. Other name for scenes in literatures is episodes or logical story units for movies², or news items in news programs. While shots are marked by physical boundaries, scenes are marked by semantic boundaries, so scene boundary detection is a far more difficult task compared with shot boundary detection. Fig 2 shows two examples of video scenes, each consists of a sequence of shots taken from the same place and in a successive order of time, and present an event.

Usually there are two steps to extract video scene structures after shot boundary detection. The first step is to represent visual content of one shot, and to define the similarity measure between two shots. The second is to construct scene structures by time-constraint shot clustering, or scene boundary detection algorithms. In most previous

works on scene extraction^{2, 3, 4, 5, 6, 7, 10}, shot representation is heavily rely on keyframe extraction and comparison, or some simple activity measure is attached to the keyframe-based shot representation scheme. Then, classical clustering algorithm or simple peak detection is used to detect scene boundaries. However, the limitation of keyframe-based shot representation is that temporal information existing in video frames is not fully exploited. Also, when a sequence of shots is considered a scene, it is often because they are semantically correlated rather than visual similarity in term of keyframes.

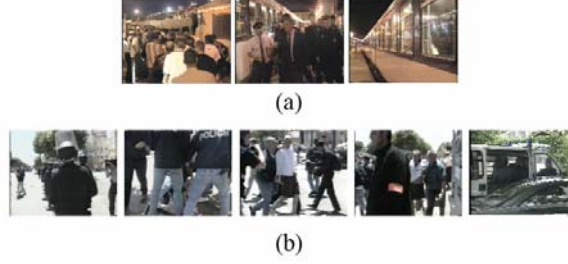


Fig.2: Two examples of video scenes, each consists of a sequence of shots taken from the same place and in a successive order of time, and present an event.

In this paper, we present a novel scheme on video content representation for shot retrieval application and a new scene boundary detection algorithm. Shots are represented with the intrinsic spatio-temporal relationships by analyzing the color and spatial content across time. Similar to¹⁰, our approach first decomposes the temporal variations of a shot into several coherent sub-units called subshots. Subshots are indispensable for describing visual content of one shot that has significant content changes, such as panning from indoor to out of window. Unlike¹⁰ that employs motion information to achieve this task, we utilize video color objects in a way that semantic content can be inherently embedded to describe the spatio-temporal changes of video content. We define a “color object” as a color sphere in HSV color space and a color-blob in the frame image. To characterize visual content variations for subshot extraction and representation, two content descriptors, Dominant Color Histogram (DCH) and Spatial Structure Histograms (SSH), are developed based on extracted color objects:

- Dominant Color Histogram (DCH) for one “group of frames”(GoF) by fusing temporal information into the frame color histograms, in order to capture the most important colors according to temporal variations;
- Spatial Structure Histograms (SSH) to represent the spatial structural information for one individual frame, and provide complementary functionality to color histograms that lack information about spatial distribution of colors.

As sub-shots being extracted, shot similarity measure can be computed for video shot retrieval based on the similarities between corresponding subshots. In addition, a new algorithm on scene boundary detection is provided by analyzing the competition of splitting and merging forces at each shot boundary.

The rest of this paper is organized as follows. In Section 2, we first introduce in detail the new shot representation, including DCH and SSH. Then we describe subshot

extraction and shot similarity measure based on DCH and SSH descriptors. Section 3 presents the scene extraction approaches based on the proposed shot representation. Experimental results are given in Section 4 and the conclusion remarks are in Section 5.

2. Shot Representation

A new shot representation scheme that exploits the intrinsic spatio-temporal relationships and contains plentiful semantics is desirable to fulfill the task of scene extraction. This is because scene is a group of shots that are semantically correlated and is a subjective concept to reflect human perception. In this section, we present a pseudo-object-based shot representation. That is, we explore the spatio-temporal relationships of a “color object”, which is defined as a color sphere in color space and a color-blob in the frame image. Color is an effective yet computational inexpensive feature, and integrating other information into color features would be preferred to improve representation capability without sacrificing efficiency. Two descriptors are introduced to describe the spatio-temporal behavior of color objects: (1) DCH that fuses temporal information into color histograms to capture the most important colors according to temporal variations; and (2) SSH that represents the spatial structural information for one individual frame, providing complementary functionality to color histograms that lack information about spatial distribution of colors. Finally, to compactly represent one shot, subshot is extracted by using DCH and SSH features, and shot similarity measure is formulated based on subshot similarity measure.

2.1 Dominant Color Histogram

Color histogram is popularly used in content-based image retrieval (CBIR) for its simplicity and effectiveness. It is natural to extend the idea of color histogram to CBVR, such as color histogram of key-frames¹¹. Another way is to construct a series of color histograms for one GoF, if every frame is represented by one color histogram. In¹², Ferman introduced one concept called alpha-trimmed average histograms to summarize the series of histograms into one single histogram, to compactly represent entire color composition in one GoF. Its basic idea is to compute mean value for one signal, after deleting some percent of extremely large or small data that are thought as abnormal. Average histogram and median histogram are special cases for alpha-trimmed average histograms, with $\alpha = 0$ and $\alpha = 0.5$, respectively. However, alpha-trimmed average histograms would be meaningless if significant variations exist in the series of histograms. In addition, this method neglects the temporal variations and visual significance for different colors.

Here we will present dominant color histogram for one GoF by dominant color extraction and tracking. In¹³, dominant color histogram for one image not only reduces the number of histogram bins but also enhances the performance of histogram matching, because it tends to capture the majority of color content and be robust to noise introduced by small bins. In this paper, dominant color histogram for one GoF depends not only on dominant colors of individual frames, but also on their temporal variations. Therefore, this representation meets the nature of video as a temporal media. Dominant color histogram is distinctive from previous work with incorporating temporal information and some semantic considerations. GoF is a general concept that may be shots, subshots, and group of shots. In the following, we will deal shot as GoF by assuming that one shot has

one single theme. If not, we can segment the shot into several subshots such that each subshot contains coherent content.

In general, we can classify shots into two types: focusing on the environment, such as a street, without dominant foreground objects; or focusing on static or moving objects, such as a car or person. The focusing background or foreground objects should have longer duration in one shot. Color is an effective yet computational inexpensive feature used in content-based retrieval. Not all the colors presented in one shot, but the dominant colors of focused objects and background, will prevail the measure of shot similarity, with considering human perception. Dominant colors should be not only dominant in one frame, but also dominant across the entire shot. We want to capture the colors of focused objects in one shot by temporal variations and weigh them according to the temporal duration. Therefore, the shot representation by dominant color histogram is emphasizing the dominant objects or background, which is very different from previous color histograms. Details on dominant color histogram are given in the following.

Firstly, we calculate the color histogram of each frame, from which dominant colors of the frame are identified. HSV color space^{1,13} is popularly used for calculating color histograms since HSV color space is natural and approximately perceptually uniform. We use the HSV color cone¹³, modified from the cylindrical HSV space, to make the differences of hue and saturation less distinguishable when illumination value is small. Also, we can define a quantization of HSV to produce a collection of colors that is compact and complete. In our method, the HSV color cone is quantized by a 3D Cartesian coordinate system with 20 values for X and Y, 10 values for Z (the lightness), respectively, as shown in Figure 3. The similarity between two colors given by indices $(h1, s1, v1)$ and $(h2, s2, v2)$ is given by the Euclidean distance between the color points $(x1, y1, z1)$ and $(x2, y2, z2)$, respectively, in the HSV color cone. The fineness of the color quantization will influent the extraction of dominant colors. A fine quantization will be able to discriminate more objects, while it may also cause the extraction of dominant colors being sensitive to lighting between frames, which may result in loss of tracking of dominant colors.

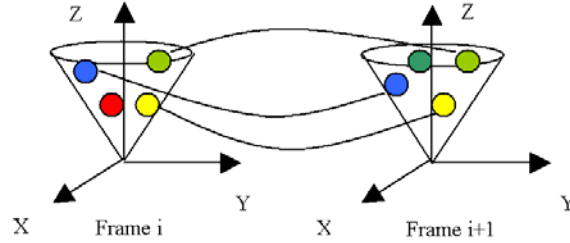


Figure 3: Dominant color extraction and tracking.

To determine dominant colors of a video shot, pixels of each frame, or DC blocks in I frames when MPEG1/2 video are used, are projected into the quantized HSV color space. The normalized distribution of these pixels in the 3D color space thus forms normalized 3D color histograms of the frame. All the dominant local maxima points in the 3D color histogram are identified; and a sphere surrounding each local maximum with a small neighborhood (with diameter of 3 quantization units) in the color space is defined as a color object. Those color objects with the largest numbers of pixels (top 20 in our implementation) are identified as dominant color objects. These dominant objects capture

the most significant color information of a frame and are more resilient to noise. We then form a 3D dominant color histogram, $hist_d(k, x, y, z)$, for each frame by counting only pixels included in dominant colors, where k denotes the frame number, and (x, y, z) denotes a color bin. We can consider pixels falling into a dominant region in the color space an object, which may (often) not represent a spatial object in a frame.

Then, dominant colors defined as above in consecutive frames are tracked in the HSV color space to identify dominant colors of a shot. If the positions of two dominant colors in two consecutive frames are sufficient close, these two colors are recognized as the same color. Such a color tracking process will continue until all frames in the shot are tracked. After tracking, only the colors that have longer duration in a shot are retained as dominant colors of an entire shot. In the words, we form an overall dominant color histogram for each shot, $hist_d^a(x, y, z)$ (a denotes a shot), consisting of only dominant colors that are not only dominant in a frame, but also dominant across the entire shot. To give more weight to colors with longer duration in a shot since they are more dominant in perception, the histogram bins, corresponding to each dominant color are weighted by its relative duration in a shot as,

$$hist_d^A(x, y, z) = hist_d^a(x, y, z) \times d_l / d_0 \quad (1)$$

where d_0 is the duration of the shot, and d_l is duration of the dominant color with color (x, y, z) . Also, $hist_d^A(x, y, z)$ is normalized by the mean size of each dominant color within the shot. Therefore, the dominant color histogram of a shot represents both structural content in a frame and temporal content in a shot. Also, these dominant colors often represent dominant objects or background in a shot and the correlation between these colors in two shots is a good representation of similarity measure between the two shots. The similarity measure between two shots, a and b , is calculated by performing the histogram intersection between two dominant color histograms of the two shots. That is,

$$Sim(a, b) = \sum_x \sum_y \sum_z \min[hist_d^A(x, y, z), hist_d^B(x, y, z)] \quad (2)$$

This similarity measure has the following nice properties:

- 1) $0 \leq Sim(a, b) \leq 1, Sim(a, a) = 1$
- 2) $Sim(a, b) = Sim(b, a)$

2.2 Spatial Structure Histogram

Spatial information would be very important to describe the global and local spatial configuration for one image. In¹⁴, a new concept called image structural feature is introduced, which is a feature in-between texture and shape but more general. Water-Filling Algorithm is proposed to extract structural features from edge maps¹⁴. In this paper, we use color-blob maps to extract a new set of features called Spatial Structure Histograms (SSH) to describe spatial information for one video frame.

Firstly, color-blob maps are obtained by color quantization (examples shown in Figure 4). Each color cluster in 3D HSV cone space is extracted by K-means clustering due to its computational simplicity and efficiency. The optimal number of clusters, k , is obtained by using the cluster separation measure

$$\rho(k) = \frac{1}{k} \sum_{i=1}^k \max_{1 \leq j \leq k, j \neq i} \left\{ \frac{\eta_i + \eta_j}{\xi_{ij}} \right\} \quad (3)$$

where η_j is the intra-cluster distance of the cluster j , and ξ_{ij} is the inter-cluster distance of cluster i and j . Note that the cluster separation measure cannot handle the case of $k=1$. The K-means algorithm is tested for $k=\{1, 2, 3, \dots, 10\}$. We choose the cluster number as 1, if the intra-cluster distance is lower than some given threshold when $k=1$. Otherwise, the cluster number is identified by the lowest value for $\rho(k)$ when $k>1$. In our implementation only the DC blocks of I frames are used, so ten color clusters would be sufficient to capture the color distribution for the general DC images. The potential problem is with texture regions that would create numerous small color regions. However, it is effectively depressed by the above cluster validity analysis that favors larger color clusters.

Several distributional features are extracted from color-blob maps, including area histogram H_{area} , position histogram H_{pos} , deviation histograms in X and Y direction, H_{vx} , H_{vy} , and span histograms in X and Y direction, H_{sx} , H_{sy} . Area histogram is computed as

$$\begin{aligned} H_{\text{area}}(i) &= \sum_{R_j \in \Omega_i} \text{Area}(R_j), i = 0, 1, \dots, 7 \\ \Omega_i &= \{R_j \mid \text{Area}(R_j) \in [A_i, A_{i+1})\}, i = 0, 1, \dots, 7 \\ A_i &= 1/2^{8-i}, i = 1, 2, \dots, 8; A_0 = 0 \end{aligned} \quad (4)$$

where $\text{Area}(R_j)$ is the area percentage of color-blob R_j . Position histogram is defined as

$$\begin{aligned} H_{\text{pos}}(i) &= \sum_{R_j \in \Omega_i} \text{Area}(R_j), i = 0, 1, 2, \dots, 15 \\ \Omega_i &= \{R_j \mid \text{Center}(R_j) \in \text{Block}(i)\}, i = 0, 1, 2, \dots, 15 \end{aligned} \quad (5)$$

where $\text{Center}(R_j)$ is the centroid of color-blob R_j , and $\text{Block}(i)$ is the i^{th} block with the image is equally divided into 16 blocks. Deviation histogram in X direction is defined as

$$\begin{aligned} H_{\text{vx}}(i) &= \sum_{R_j \in \Omega_i} \text{Area}(R_j), i = 0, 1, 2, \dots, 7 \\ \Omega_i &= \{R_j \mid \sigma_x(R_j) \in [B_i, B_{i+1})\}, i = 0, 1, 2, \dots, 7 \\ B_i &= 1/2^{8-i}, i = 1, \dots, 8; B_0 = 0 \end{aligned} \quad (6)$$

where $\sigma_x(R_j)$ is the standard deviation of color-blob R_j in x direction, normalized by the image width. H_{vy} is similarly defined except that $\sigma_y(R_j)$ is normalized by image height. Span histogram in X direction is defined as

$$\begin{aligned} H_{\text{sx}}(i) &= \sum_{R_j \in \Omega_i} \text{Area}(R_j), i = 0, 1, 2, \dots, 7 \\ \Omega_i &= \{R_j \mid \text{Width}(R_j) \in [B_i, B_{i+1})\}, i = 0, 1, 2, \dots, 7 \\ B_i &= 1/2^{8-i}, i = 1, \dots, 8; B_0 = 0 \end{aligned} \quad (7)$$

where $Width(R_j)$ is the width of minimum bounding rectangle (MBR) of color-blob R_j , normalized by the image width. H_{sy} is similarly defined except that $Height(R_j)$ is normalized by image height.

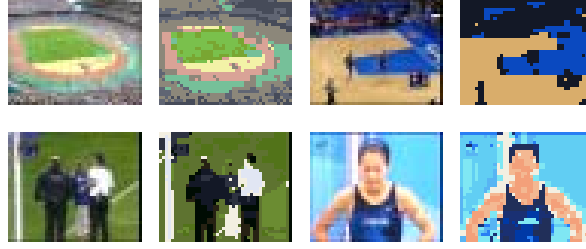


Figure 4: Examples of segmented color-blob maps.

Area histogram describes the spatial complexity of the image. Position histogram is desired to identify similar spatial configuration such as a close-up shot on the head and shoulder of one player. Deviation histograms and span histograms represent the shape distributions of the color-blob map. We found that the span histograms are not rotation-invariant and not as robust as others, so span histograms are not used in our experiments. The spatial similarity between two images, a and b , is computed as

$$SshSim(a, b) = W0 \times Sim0 + W1 \times Sim1 + W2 \times Sim2 + W3 \times Sim3 \quad (8)$$

where $Sim0$, $Sim1$, $Sim2$, and $Sim3$ are histogram similarity on H_{area} , H_{pos} , H_{vx} , and H_{vy} , respectively, by using histogram intersection. $W0$, $W1$, $W2$, and $W3$ are corresponding weights that are equally set in our experiments.

2.3 Subshots extraction and shot similarity measure

It would be better to segment one shot with significant content variations into several subshots, because the aggregated representation is unpredictable if we compose all the variations into one feature vector, such as for one shot panning from indoor to outdoor. We propose one simple subshot extraction algorithm based on color and spatial structure changes, because subshot should be of coherent visual content for compact representation. Suppose the percentage of newly emerged dominant color bins is $P1$, the difference of spatial structural histogram between the current and previous I frame is $P2$. A new subshot is identified if all the following conditions are true:

$$P1 > T1, \text{ and } P2 > T2, \text{ and } P1 + P2 > T3 \quad (9)$$

where $T1$, $T2$, and $T3$ are predefined threshold (in our experiment they are empirically set as 0.2, 0.2, 0.6). The similarity measure of two shots, a and b , is defined as

$$Sim(a, b) = \max_{i, j} (Sim(a_i, b_j)) \quad (10)$$

where $Sim(a_i, b_j)$ is the similarity of the subshot i in shot a and the subshot j in shot b , which can be computed in two ways:

$$Sim(a_i, b_j) = Wc \times DchSim(a_i, b_j) + Ws \times SshSim(a_i, b_j) \quad (11)$$

$DchSim()$ is the similarity on DCH of two subshots, and $SshSim()$ is the similarity on average SSH for two subshots. Wc and Ws are the corresponding weights that are equally set in our experiment.

3. Scene Boundary Detection

With the similarity measure between two shots defined above, we have developed a video scene boundary extraction approach based on a competition analysis of splitting and merging forces. That is, each shot is subject to two kinds of forces from its neighborhood. One is the splitting force in two directions, from the previous shots and the following shots, respectively. If the ratio of splitting forces in two directions is large, there could be a potential scene boundary. Another is the merging force between the previous and following shots, preventing the current shot being attracted by one side and trying to merge the previous and following shots into the same scene. A simple rule-based algorithm is used to detect the signal jumps on the two forces.

Considering the temporal constraints, i.e. shots that are closer to each other in time are more likely to belong to the same scene, the similarity score between two shots is weighted by temporal attraction factor:

$$w = 1/(1+d/C) \quad (12)$$

where d is the time span between the two shots (from the ending frame of the previous shot to the beginning frame of the current shot), and C is a constant (20 seconds in our implementation).

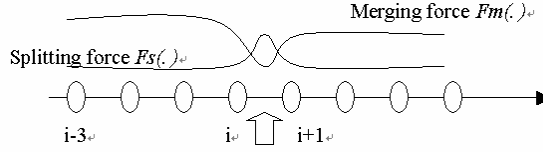


Fig 5: Scene boundary detection by force competition.

We define an splitting force to shot i from the previous shots as

$$Fs(i) = left(i) / right(i) \quad (13)$$

where

$$\begin{aligned} left(i) &= \max\{sim(i, i-1), sim(i, i-2), sim(i, i-3)\} \\ right(i) &= \max\{sim(i, i+1), sim(i, i+2), sim(i, i+3)\} \end{aligned}$$

The splitting force for current scene boundary candidate $(i|i+1)$ is defined as

$$Fs(i|i+1) = (Fs(i)+1/Fs(i+1))/2 \quad (14)$$

The physical meaning of splitting force $F_s(i|i+1)$ is the average of splitting force to shot i from previous shots and splitting force to shot $i+1$ from following shots.

The merging force for shot i is defined as

$$F_m(i) = \frac{1}{3} \sum_{j=i-3}^{i-1} \max(\text{sim}(j, i+1), \text{sim}(j, i+2), \text{sim}(j, i+3)) \quad (15)$$

Its physical meaning is the average attraction between previous shots and following shots to shot i . The merging force for current scene boundary candidate $(i|i+1)$ is defined as

$$F_m(i|i+1) = (F_m(i) + F_m(i+1))/2 \quad (16)$$

Then we normalize $F_s(i|i+1)$ and $F_m(i|i+1)$ to $F_s'(i|i+1)$ and $F_m'(i|i+1)$ with following linear scaling to unit variance:

$$x' = \frac{\frac{x-\mu}{3\sigma} + 1}{2} \quad (17)$$

It is guaranteed that 99% of $F_s'(i|i+1)$ and $F_m'(i|i+1)$ is in $[0,1]$.

At an ideal scene boundary, $(i|i+1)$, $F_s'(i|i+1)$ should reach its maximum and $F_m'(i|i+1)$ should reach its minimum. However, it is not always this case. Therefore, these two situations are treated differently as following:

Condition 1: $F_s'(i|i+1)$ reaches its maximum and $F_m'(i|i+1)$ reaches its minimum simultaneously. A scene boundary is identified if $F_s'(i|i+1) > T1$;

Condition 2: $F_s'(i|i+1)$ reaches its maximum, or $F_m'(i|i+1)$ reaches its minimum. If $F_s'(i|i+1) > T2$ and $F_s'(i|i+1) - F_m'(i|i+1) > T3$, a scene boundary is declared.

where $T1$, $T2$, and $T3$ are predefined thresholds (set as 0.6, 0.7, 0.2, respectively, in our experiments) and the neighborhood is set as $[-2,2]$ to detect maximum and minimum.

4. Experimental Results

4.1 Shot retrieval

The experiment was conducted on 815 shots from a dozen of TV sports programs, with totally 163880 frames (110 minutes). The video database is very challenging because it contains a diversity of sports programs, including ball games, track and field events, diving, boxing, etc. Figure 6 shows the user interface of our experiments. We chose 8 semantic classes of querying shots as basketball-court, tennis-court, diving, swimming, close-up on head and shoulder, audience, underwater, and stadium, with examples shown in Figure 7. For each class we randomly picked out 20 shots as query examples. Eight retrieval methods were tested for compare:

- I. Average Color Histogram (ACH);
- II. Median Color Histogram (MCH);
- III. Dominant Color Histogram (DCH);
- IV. Spatial Structure Histograms (SSH);
- V. Weighted sum of DCH and SSH similarities;
- VI. Product of DCH and SSH similarities;
- VII. Weighted sum of DCH and SSH similarities for subshots;
- VIII. Product of DCH and SSH similarities for subshots.

We adopted the average normalized modified retrieval rank (ANMRR) in MPEG-7 standardization activities^{11,7} and the average recall (AR) as performance measures. A low value of ANMRR means the relevant shots ranked at the top, and a high value of AR denotes more relevant shots found in the top K (K is the cut-off number for retrieval). Experimental results on AR and ANMRR for different shot classes (from 0 to 7) with different methods (from I to VIII) are shown in Table 1 and 2. Several query results are shown in Fig 8-13. The overall performances are on par, except that of SSH because sports video has a wide range of spatial structure variations than color components. Within color histogram methods I, II, and III, the results of DCH is slightly better than ACH and MCH with considering that DCH achieves the best AR but almost the same ANMRR with ACH and MCH. The performance difference between DCH and the other two is supposed to be greater for the object-tracking shots with significant content variations. For methods V, VI, VII and VIII, the equally weighted sum of DCH and SSH similarities is inferior to the product versions, showing that weighted sum is not a good fusing approach. Overall, method VIII outperforms all the others to show that subshot extraction and representation by DCH and SSH could achieve the best performance on ANMRR and AR simultaneously.

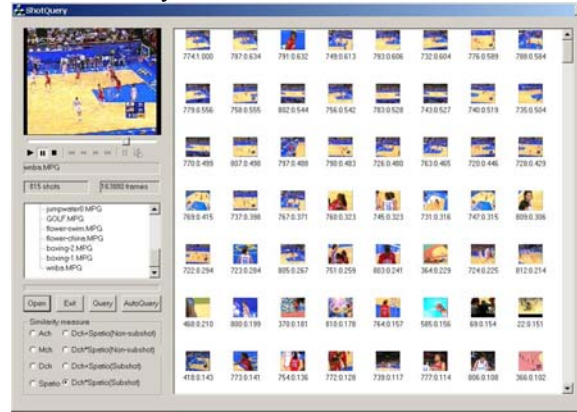


Figure 6: Interface for the shot retrieval experiments.



Figure 7: Shot examples of the semantic classes.

Table 1: AR for different shot classes (from 0 to 7) with different methods (from I to VIII).

	I	II	III	IV	V	VI	VII	VIII
0	0.3951	0.3996	0.3916	0.3637	0.3925	0.4018	0.3850	0.3965
1	0.4417	0.4514	0.4556	0.2694	0.4722	0.4889	0.4708	0.4792
2	0.4967	0.4967	0.5025	0.5475	0.5558	0.5125	0.5667	0.5150
3	0.8485	0.8515	0.8758	0.4727	0.7742	0.8894	0.7682	0.8939

4	0.4737	0.5158	0.5000	0.2553	0.4053	0.5000	0.4211	0.4947
5	0.8227	0.7818	0.7818	0.6409	0.7818	0.7909	0.8318	0.8636
6	0.7387	0.7403	0.8016	0.3177	0.7145	0.7919	0.7129	0.7823
7	1.0000	1.0000	1.0000	0.7000	1.0000	1.0000	1.0000	1.0000
AR	<i>0.7453</i>	<i>0.7482</i>	<i>0.7584</i>	<i>0.5096</i>	<i>0.7281</i>	<i>0.7679</i>	<i>0.7366</i>	<u><i>0.7750</i></u>

Table 2: ANMRR for different shot classes (from 0 to 7) with different methods (from I to VIII).

	I	II	III	IV	V	VI	VII	VIII
0	0.6919	0.7046	0.7093	0.7405	0.7066	0.7031	0.7097	0.7075
1	0.6925	0.6780	0.6930	0.8145	0.6882	0.6810	0.6843	0.6801
2	0.4691	0.4724	0.4690	0.5687	0.4558	0.4711	0.4502	0.4696
3	0.3023	0.2970	0.2932	0.6853	0.3477	0.2629	0.3274	0.2516
4	0.6276	0.5903	0.6163	0.8230	0.6699	0.6214	0.6620	0.6212
5	0.2526	0.2514	0.2706	0.4552	0.2521	0.2539	0.2650	0.2620
6	0.3361	0.3401	0.2945	0.7630	0.3590	0.3000	0.3517	0.3029
7	0.0142	0.0108	0.0128	0.4357	0.0057	0.0037	0.0060	0.0045
ANMRR	<i>0.4838</i>	<i>0.4778</i>	<i>0.4800</i>	<i>0.7552</i>	<i>0.4978</i>	<u><i>0.4710</i></u>	<i>0.4937</i>	<u><i>0.4714</i></u>

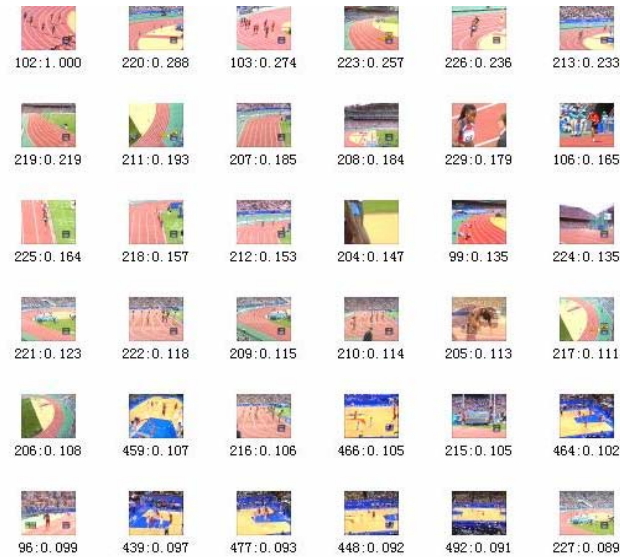


Fig 8: Query results for an arena shot.

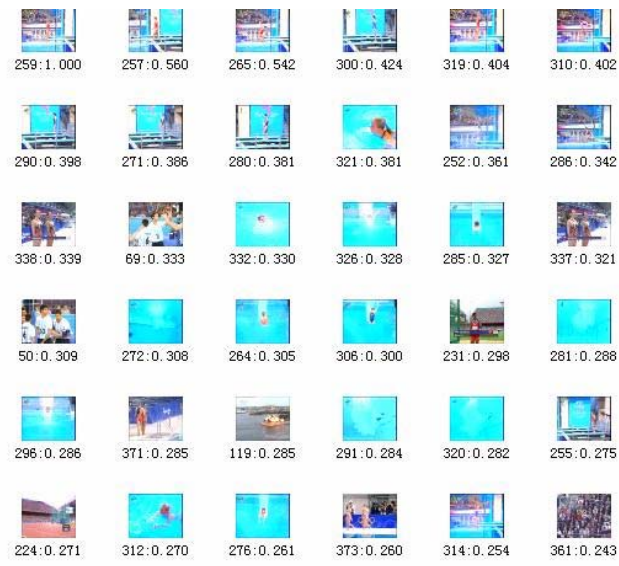


Fig 9: Query results for a diving shot.

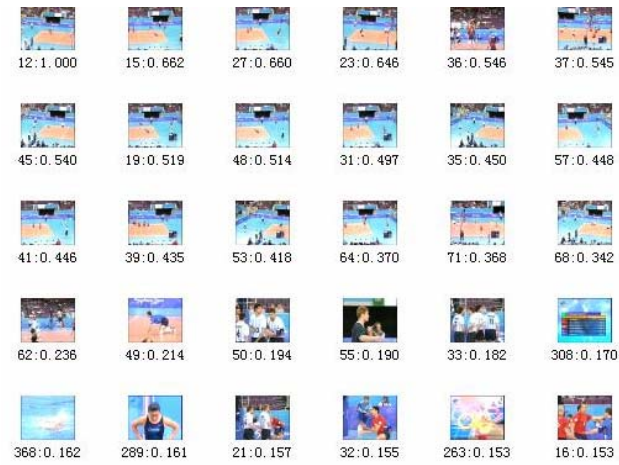


Fig 10: Query results for a volleyball game shot.



Fig 11: Query results for a close-up shot.

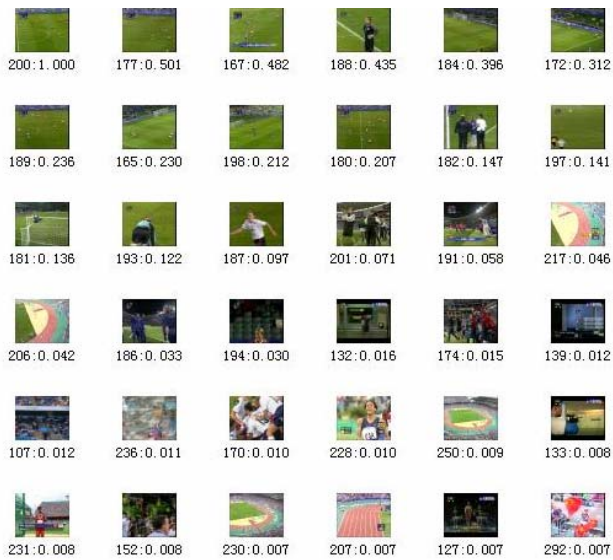


Fig 12: Query results for a soccer shot.



Fig 13: Query results for a shot on multiple players.

4.2 Scene Extraction

Two MPEG-7 test videos are used to evaluate our algorithm on scene boundary detection: *lgerca_lisa_1.mpg* and *lgerca_lisa_2.mpg*. They are both home videos and each has approximately 32,000 frames. Experimental results are listed in Table 3 and 4. Figure 14 and 15 show the splitting and merging forces for each video, and several scene examples are shown in Fig 16-19. In most cases, splitting force reaches its maxima and merging force reaches its minima simultaneously at one ground-truth scene boundary. Three false scene boundaries in *lgerca_lisa_1.mpg* are caused by background differences and lighting condition variations. In *lgerca_lisa_2.mpg* five scene boundaries are missed, but the results are reasonable when considering that scene is a subjective concept and different human subject has different rules to extract scenes. The original ground-truth scene boundaries defined by source provider are arguable because scene 1 to 5 are taken in gym, scene 9 to 11 are on stage, and scene 13 to 14 are in swimming pool. In summary, the results are very promising compared with other approaches.

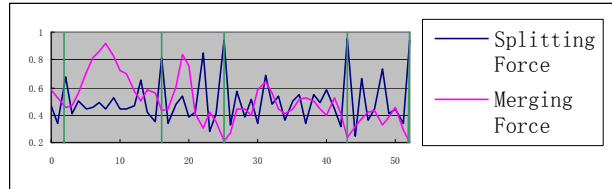


Figure 14: Force competition for *lgerca_lisa_1.mpg*, with ground-truth scene boundaries marked by a vertical line.

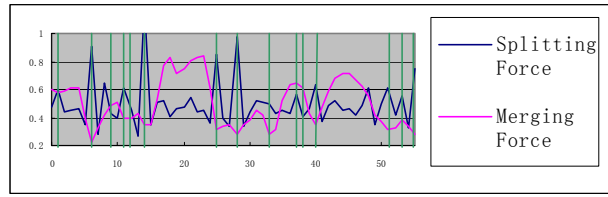


Figure 15: Force competition for *lgerca_lisa_2.mpg*, with ground-truth scene boundaries marked by a vertical line.



Fig 16: Scene 3 (hot balloon event) for *lgerca_lisa_1.mpg*.



Fig 17: Scene 4 (playing on lawn) for *lgerca_lisa_1.mpg*.



Fig 18: Scene 8 (kids driving) for *lgerca_lisa_2.mpg*.



Fig 19: Scene 12 (after play) for *lgerca_lisa_2.mpg*

Table 3: Scene boundaries for *lgerca_lisa_1.mpg*

Scene	Scene Description	Shots	Correct	Miss	False
0	Kids learning roller-skater	0-1	1 2		
1	Kids playing in gym	2-15	15 16		12 13
2	Kid playing with water with parents	16-24	24 25		21 22
3	Hot balloon event	25-42	42 43		
4	Kids playing with parent on lawn	43-51	51 52		47 48

5	Indoor exercise	52	\	\	\
---	-----------------	----	---	---	---

Table 4: Scene boundaries for *lgerca_lisa_2.mpg*.

Scene	Scene Description	Shots	Correct	Miss	False
0	Kids at home with cat	0	0 1		
1	Kids in gym	1-5	5 6		
2	Two kids playing high-bar (Over-illuminated)	6-8		8 9	7 8
3	Kid + teacher	9-10	10 11		
4	Kids jumping	11		11 12	
5	Kids in Gym	12-13	13 14		
6	Kids playing games in Gym (Over-illuminated)	14-24	24 25		
7	Kids playing at home (Dim lighting)	25-27	27 28		
8	Kids driving outside home	28-32	32 33		
9	Kids dancing on stage (Part I of Play)	33-36		36 37	
10	(Part II of Play)	37		37 38	
11	(Part III of Play)	38-39	39 40		
12	After Play	40-50	50 51		
13	Swimming Pool	51-52		52 53	
14	Crowded Swim Pool	53-54	54 55		
15	Kid's party	55	\	\	\

5. Conclusions

In this paper, we present a novel scheme on shot content representation and similarity measure by subshots extraction and representation. Two content descriptors are developed to measure video content variations and represent subshots: dominant color histogram (DCH) and spatial structure histograms (SSH). Eight similarity measures are tested and our proposed method (subshot representation by DCH and SSH) gives the best performance on ANMRR and AR. Also, we have presented a new scene boundary detection algorithm by splitting and merging force competition. Promising results are achieved by our scene extraction algorithm for MPEG-7 test videos. Our future work is to develop new motion descriptors that could be incorporated in the subshot representation scheme with DCH and SSH to improve the performance of shot retrieval applications and the scene extraction algorithm.

6. References

1. Y. Rui and T. S. Huang, A Uniform Framework for Video Browsing and Retrieval, *The Image and Video Processing Handbook*, Alan Bovik, ed., Academic Press, 2000. A. Hanjalic, R. L. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced video-retrieval systems", *IEEE Transactions on Circuits and Systems For Video Technology*, Vol. 9, No. 4, pp. 580-588, June 1999.
2. J. M. Corridoni and A. Del Bimbo, "Structured representation and automatic indexing of movie information content", *Pattern Recognition*, Vol. 31, No. 12, pp. 2027-2045, 1998.
3. B. Gunsel, Y. Fu, and A. M. Tekalp, "Hierarchical temporal video segmentation and content characterization", in *Multimedia Storage and Archiving Systems II*, Proc. SPIE Vol. 3229, pp. 46-56, 1997.
4. Y. Rui, T. S. Huang, and S. Mehrotra, "Exploring video structure beyond the shots", Proc. IEEE Conf. on Multimedia Computing and Systems, pp. 237-240, 1998.
5. J. R. Kender and B. L. Yeo, "Video scene segmentation via continuous video coherence", *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 367-373, 1998.
6. J. Huang, Z. Liu and Y. Wang, "Integration of Audio and Visual Information for Content-based Video Segmentation", *Proc. ICIP'98*, Chicago, Oct. 1998.
7. L. Zhao, et al, "Key-frame extraction and shot retrieval using Nearest Feature Line (NFL)", *International Workshop on Multimedia Information Retrieval*, in conjunction with *ACM Multimedia2000*, Los Angeles, USA, 2000.
8. S. Z. Li, K. L. Chan, and C. L. Wang, "Performance evaluation of the Nearest Feature Line method in image classification and retrieval", *IEEE Trans. on PAMI*, Vol. 22, No. 11, Nov. 2000.
9. C. W. Ngo, T. C. Pong, H. J. Zhang, and R. T. Chin, "Motion-based video representation for scene change detection", *ICPR'00*, 2000.
10. H. J. Zhang et al, "An integrated system for content-based video retrieval and browsing", *Pattern Recognition*, May 1997.
11. A. M. Ferman, et al, "Group-of-frames/pictures color histogram descriptors for multimedia applications", *ICIP2000*.
12. W. Y. Ma and H. J. Zhang, "Content-based image indexing and retrieval", in *Handbook of Multimedia Computing*, Borko Furht, ed. CRC Press, 1998.
13. X. S. Zhou, Y. Rui, and T. S. Huang, "Water-Filling: a novel way for image structural feature extraction", *ICIP'99*.
14. MPEG Video Group, Description of core experiments for MPEG-7 color/texture descriptors, *ISO/MPEGJTC1/SC29/ WG11 MPEG98/M2819*, July 1999.
15. T. Lin, H. J. Zhang, "Automatic video scene extraction by shot grouping", *ICPR'00*, 2000.
16. C. W. Ngo, T. C. Pong, H. J. Zhang, and R. T. Chin, "Motion characterization by temporal slices analysis", *Proceedings of CVPR2000*, vol. 2, pp. 768-773, 2000.
17. A. Hanjalic, R. L. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced video-retrieval systems", *IEEE Transactions on Circuits and Systems For Video Technology*, Vol. 9, No. 4, pp. 580-588, June 1999.
18. J. M. Corridoni and A. Del Bimbo, "Structured representation and automatic indexing of movie information content", *Pattern Recognition*, Vol. 31, No. 12, pp. 2027-2045, 1998.
19. Y. Rui, T. S. Huang, and S. Mehrotra, "Exploring video structure beyond the shots", *Proc. IEEE Conf. on Multimedia Computing and Systems*, pp. 237-240, 1998.
20. J. R. Kender and B. L. Yeo, "Video scene segmentation via continuous video coherence", *Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pp. 367-373, 1998.
21. A. M. Ferman, et al, "Group-of-frames/pictures color histogram descriptors for multimedia applications", *ICIP'00*, 2000.
22. X. S. Zhou, Y. Rui, and T. S. Huang, "Water-Filling: a novel way for image structural feature extraction", *ICIP'99*.



Tong Lin received the BS degree from Chang-Chun Normal College, China, in 1996. He is currently finishing the Ph.D. degree in National Laboratory on Machine Perception at Peking University, China. In 1999 and 2000, he was a summer research intern at the Media Computing Group of Microsoft Research, China. His current research interests include multimedia information retrieval, pattern recognition, and multimedia applications.



HongJiang Zhang received his Ph.D from the Technical University of Denmark and his BS from Zhengzhou University, China, both in Electrical Engineering, in 1982 and 1991, respectively.

From 1992 to 1995, he was with the Institute of Systems Science, National University of Singapore, where he led several projects in video and image content analysis and retrieval and computer vision. He also worked at MIT Media Lab in 1994 as a visiting researcher. From 1995 to 1999, he was a research manager at Hewlett-Packard Labs, where he was responsible for research and technology transfers in the areas of multimedia management; intelligent image processing and Internet media. In 1999, he joined Microsoft Research, China, where he is currently a Senior Researcher and the Assistant Managing Director mainly in charge of media computing and information processing research.

Dr. Zhang is a Senior Member of IEEE and a member of ACM. He has authored 3 books, over 120 referred papers and book chapters, 7 special issues of international journals in multimedia processing, content-based media retrieval, and Internet media, as well as numerous patents or pending applications. He currently serves on the editorial boards of five professional journals and a dozen committees of international conferences.



Qing-Yun Shi is a professor of National Laboratory on Machine Perception, Peking University and a member of the Chinese Academy of Sciences. She was a member of the Governing Board of International Association for Pattern Recognition from 1990 to 2000. Her current research interests include multimedia information processing, image databases, and biometrics.