Proceedings of the 5th WSEAS Int. Conf. on MULTIMEDIA, INTERNET AND VIDEO TECHNOLOGIES, Corfu, Greece, August 17-19, 2005 (pp212-217)

On the Query of Video Database

Liang-Hua Chen and Kuo-Hao Chin
Department of Computer Science
and Information Engineering
Fu Jen University, Taipei, Taiwan

Hong-Yuan Mark Liao
Institute of Information Science
Academia Sinica
Taipei, Taiwan

Abstract: The usefulness of a video database depends on whether the video of interest can be easily located. In this paper, we propose a video retrieval algorithm based on a new representation of video shot. In contrast to key-frame based representation of shot, our approach analyzes all frames within a shot to fully exploit the spatio-temporal information contained in video. A compact shot representation is obtained by integrating the color and spatial features of individual frame. In the video matching step, a shot similarity measure is defined to locate the occurrence of similar video clips in the database. Experimental results indicate that the proposed approach is effective and feasible in retrieving and ranking similar video clips.

Keywords: Video retrieval, feature extraction, similarity measure.

1. Introduction

The advances in low cost mass storage devices, higher transmission rates and improved compression techniques, have led to the widespread use and availability of digital video. Video data offers users of multimedia systems a wealth of information and also serves as a data source in many applications including digital libraries, publishing, entertainment, broadcasting and education. The usefulness of these applications depends largely on whether the video of interest can be retrieved accurately within a reasonable amount of time. Video query by keywords is inefficient, because it is not easy to describe video content in words. Alternatively, query-by-example is a more feasible approach that searches video database according to the visual content of query example.

The query example may be an image, a shot or a clip. A shot is a sequence of video frames that was continuously captured by the same camera, while a clip is a series of shots describing a particular event. For example, a dialogue clip between two people may have a shot of speaker A, followed by a shot of the other speaker B, followed by a wide-angle shot of two parties involved. Video retrieval based on a single shot may not be practical since a shot itself is only a part of an event and does not convey full story. On the other hand, clip-based retrieval is more concise and convenient for most casual users. Thus, our problem can be formulated as: given a sample clip, find all occurrences of similar (or relevant) video clips in the database.

Current techniques for content-based video retrieval can be broadly classified into two categories: frame sequence matching[1, 2, 3] and key-frame based shot matching[4, 5, 6, 7]. The first one is derived from the sequential correlation matching widely used in the signal processing domain. These methods usually focus on frame-by-frame comparison between two clips in order to find sequences of frames that are consistently similar. The common drawback of these techniques is the heavy computational cost of the exhaustive search. Although there exist some techniques[8, 9] to improve the linear scanning speed, their time complexity still remains at least linear to the size of database. Additionally, these approaches are susceptible to alignment problem when comparing clips of different encoding rates. In the second category, each video shot is represented by a key-frame compactly. To reduce computational cost, video sequence matching is achieved by comparing the visual features of key-frames. The problem with these approaches lies in that they all leave out the temporal variations and correlation between key-frames within an individual shot. Also, it is not clear

as to which image should be used as the key-frame for a shot. To strike a good balance between searching accuracy and computational cost, in this paper, we propose a new shot content representation for shot matching. In contrast to previous approaches, our approach analyzes all frames within a shot to fully exploit the spatio-temporal information contained in video.

The main issues regarding content-based video retrieval are: (1) how to select visual features to represent the content of a video clip and (2) how to define a distance metric to measure the visual similarity between two video clips. The next section of this paper describes the visual features used in our work. Then, the proposed shot similarity measure and video matching algorithm are described in Section 3. The performance evaluation of our approach is reported in Section 4. Finally, some concluding remark is given in Section 5.

2. Visual Feature

Shot is the fundamental unit of a video. To facilitate subsequent video analysis, in our system, the query video clip and database video are segmented into shots. This task is achieved by applying our shot boundary detection algorithm [10] to the original video sequence. The next issue is the compact representation of video content for shot similarity measure and retrieval.

Color is one of the most widely used visual features in video content analysis, because it is an important source of information in visual content for discrimination. However, the amount of color information in video is vast. The raw data of video has to be transformed into compact feature representation that conveys only the most salient color aspects of the visual content. Color histogram is the most the most commonly used color feature representation. The histogram-based approach is relatively simple to calculate and can provide reasonable results. However, due to the statistical nature, color histogram does not capture spatial layout information of each color. When the image collection is large, two different content images are likely to have quite similar histograms. To remedy this deficiency, the distribution state of each single color in the spatial (image) domain needs to be taken into account.

The color histogram for an image is constructed by counting the number of pixels of each color. The main issues regarding the construction of color histogram involve the choice of color space and quantization of color space. The RGB color space is the most common color format for digital images, but it is not perceptually uniform. Uniform quantization of RGB space gives perceptually redundant bins and perceptual holes in color space. Therefore, the non-uniform quantization may be needed. Alternatively, HSV (hue, saturation,intensity) color space is chosen since it is nearly perceptually uniform. Thus, the similarity between two colors is determined by their proximity in the HSV color space. When a perceptually uniform color space is chosen , uniform quantization may be appropriate. Since the human visual system is more sensitive to the hue than to the saturation and the value [11], H should be quantized finer than S and V. In our implementation, hue is quantized into 20 bins. Saturation and intensity are each quantized into 10 bins. This quantization provides 2000 ($= 20 \times 10 \times 10$) distinct colors (bins), and each bin with non-zero count corresponds to a *color object*.

Since we are interested in the whole shot rather than single image frame, only one histogram is used to count the color distribution of all image frames within a shot. Then, each bin of the resulting histogram is divided by the frame number of a shot to obtain the average histogram. Next, several spatial features are calculated to characterize the distribution state of each color object in each image frame. Assuming a set of pixels $S = \{(x_1, y_1), \cdots, (x_n, y_n)\}$ belong to color object c_i, k is the image size and m is the total number of 4-connected pixels in S. Then, we define

(i) density of distribution

$$f_{i1} = \frac{n}{k}$$

(ii) compact of distribution

$$f_{i2} = \frac{m}{n}$$

(iii) scatter

$$f_{i3} = \frac{1}{n\sqrt{k}} \sum_{j=1}^{n} \sqrt{(x_j - x_\mu)^2 + (y_j - y_\mu)^2}$$

where $x_\mu = \frac{1}{n} \sum_{i=1}^{n} x_i$ and $y_\mu = \frac{1}{n} \sum_{i=1}^{n} y_i$

To define the fourth feature, the image is equally partitioned into p blocks of size 16×16. A block is *active*, if it contains some subset of S. Let the number of active blocks in the image frame be q, we define
(iv) ratio of active block

$$f_{i4} = \frac{q}{p}$$

After the spatial features of all image frames within a shot are computed, we take average of these values respectively. Let $\overline{f_{i1}}, \overline{f_{i2}}, \overline{f_{i3}}$ and $\overline{f_{i4}}$ be the average feature values of color object c_i within a shot, for two color objects c_i and c_j, the difference of spatial distribution within a shot is defined as

$$\begin{aligned} D_s(c_i, c_j) &= w_1|\overline{f_{i1}} - \overline{f_{j1}}| + w_2|\overline{f_{i2}} - \overline{f_{j2}}| + \\ & \quad w_3|\overline{f_{i3}} - \overline{f_{j3}}| + w_4|\overline{f_{i4}} - \overline{f_{j4}}| \quad (1) \end{aligned}$$

In our experiment, we set $w_1 = w_2 = w_3 = w_4 = \frac{1}{4}$.

3. Video Matching

Our approach performs video matching at two levels. At the shot level, the objective is to evaluate the visual similarity between two shots with different durations (lengths). At the sequence level, the video matching is achieved by sliding the query video clip (a matching window) along the database video at one shot increment and computing the similarity metric for every window position.

Let A,B be the set of all color objects in shot S_1 and S_2 respectively, for a given $u \in A$, its similar color object in B is some $v \in B$ such that $||u - v|| < \epsilon$, where $||u - v||$ denotes the Euclidean distance between u and v in the HSV color space and ϵ is a threshold. Then, (u, v) is called a *similar color pair*. Let $\Omega = \{(u, v)|(u, v) \in A \times B, (u, v)$ is a similar color pair$\}$, the shot similarity measure between S_1 (with the average histogram $\overline{H_1}$) and S_2 (with the average histogram $\overline{H_2}$) is defined as

$$\begin{aligned} \text{ShotSim}(S_1, S_2) &= \frac{1}{k} \sum_{(u,v)\in\Omega} \{W(D_s(u, v)) \times \\ & \quad \min(\overline{H_1}(u), \overline{H_2}(v))\} \quad (2) \end{aligned}$$

where k is the image size, D_s is the difference of spatial features as defined in equation (1) and W is a weight function defined as

$$W(D) = \frac{0.2}{1 + e^{10\times D-5}} + 0.8$$

It is used to fuse spatial distribution information with histogram. The construction of this weight function is motivated by the psychophysical observation: the effect of spatial distribution on human perception is progressive[12]. Only when the difference in spatial features is greater than a threshold, human perceive significant visual variation.

It is noted that a given color object in shot S_1 may have more than one similar color objects in shot S_2 as illustrated in Fig. 1. To avoid the overlapping contribution in calculating shot similarity, after each step of $\min(\overline{H_1}(u), \overline{H_2}(v))$, $\overline{H_1}(u)$ and $\overline{H_2}(v)$ are all subtracted by $\min(\overline{H_1}(u), \overline{H_2}(v))$.

Given the query video clip $Q = \{q_1, \cdots, q_m\}$ and the database video $V = \{v_1, \cdots, v_n\}$, where q_i and v_j denote the segmented shots, the similarity measure between the query clip and the database video segment starting at the i-th shot is defined as

$$D_i = \sum_{j=1}^{m} \text{ShotSim}(q_j, v_{i+j-1}) \quad (3)$$

If D_i is a local maxima and is also greater than a threshold T then a similar clip is detected at the i-th shot of database video.

4. Experimental Results

To evaluate the performance of the proposed approach, we set up a database that consists of 10 hours of videos approximately. The genres of videos include news, sports, movies and documentaries. The testing with different genres of videos would ensure that the overall performance of the algorithm is not biased toward a specific video category. Fig. 2 shows an example of retrieving and ranking similar video clips with query clip (shown in the first row). In each row, sampled frames (one for each shot) are used to represent the content of video clip. As shown in Fig. 2, the retrieved results are similar to the query clip, and they are ranked in descending order of similarity. Fig. 3 shows another example of video retrieval.

The performance of video retrieval is usually measured by the following two metrics:

$$\text{Recall} = \frac{DC}{DB} \qquad \text{Precision} = \frac{DC}{DT}$$

where DC is the number of similar clips which are detected correctly, DB is the number of similar clips in the database and DT is the total number of detected clips. The ground truth of database, i.e., the decision whether a video clip is similar or not, is determined by human subjects. Table 1 gives the experimental results using five different queries.

Table 1: Performance of video retrieval.

#	Content	Recall	Precision
1	Close-up interview shots	0.67	0.86
2	Hot-air Balloons	0.50	0.60
3	Scene of a male character	0.88	0.78
4	Free throw shots	0.71	0.83
5	Classroom scene	0.71	0.63

5. Conclusion

We have presented a new video shot representation and a video similarity measure to achieve video retrieval task. To fully exploit the spatio-temporal information contained in video, our approach analyzes all frames within a shot. A single representation for the entire shot is obtained by integrating the color and spatial features of individual frame. The system is able to locate similar video sequences encoded at different frame rates. Experimental results indicate that the proposed approach is effective and feasible in retrieving and ranking similar video clips. Finally, our future work should incorporate other video features, such as audio and text, for assessing video similarity.

References

[1] R. Mohan. Video sequence matching. In *Proceedings of International Conference on Acoustic, Speech and Signal Processing*, pages 3697–3700, 1998.

[2] Y.P. Tan, S.R. Kulkarni, and P.J. Ramadge. A framework for measuring video similarity and its application to video query by example. In *International Conference on Image Processing*, pages 106–110, October 1999.

[3] M.R. Naphade, M.M. Yeung, and B.L. Yeo. A novel scheme for fast and efficient video sequence matching using compact signature. In *SPIE Conference on Storage and Retrieval for Media Database*, pages 564–572, January 2000.

[4] R. Lienhart, W. Effelsberg, and R. Jain. VisualGREP: A systematic method to compare and retrieve video sequences. In *SPIE Conference on Storage and Retrieval for Image and Video Database*, pages 271–282, January 1998.

[5] A.K. Jain, A. Vailaya, and X. Wei. Query by video clip. *Multimedia Systems*, 7:369–384, 1999.

[6] X. Liu, Y. Zhung, and Y. Pan. A new approach to retrieve video by example video clip. In *ACM International Conference on Multimedia*, pages 41–44, 1999.

[7] Y. Peng and C.-W. Ngo. Clip-based similarity measure for hierarchical video retrieval. In *International Workshop on Multimedia Information Retrieval*, pages 53–60, October 2004.

[8] K. Kashino, T. Kurozumi, and H. Murase. A quick search method for audio and video signals based on histogram pruning. *IEEE Transactions on Multimedia*, 5(3):348–357, September 2003.

[9] J. Yuan, Q. Tian, and S. Ranganath. Fast and robust search method for short video clips from large video collection. In *International Conference on Pattern Recognition*, pages 866–869, August 2004.

[10] L.-H. Chen, C.-W. Su, H.-Y. Liao, and C.-C. Shih. On the preview of digital movies. *Journal of Visual Communication and Image Representation*, 14(3):357–367, September 2003.

[11] X. Wan and C.-C. Jay Kuo. A new approach to image retrieval with hierarchical color clustering. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):628–643, September 1998.

[12] C.S. Li et. al. *Vision and Cognition*. Yuan Liou Publishing, Taipei, Taiwan, 1999.

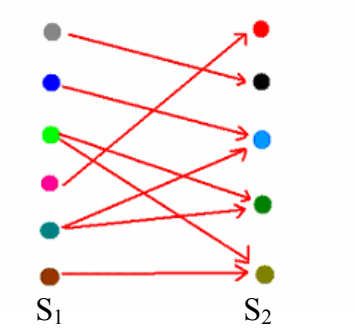

Fig. 1: Finding similar color object pairs

Fig. 2: Retrieval result for a free throw query.

Fig. 3: Retrieval result for a male character query.