Fast Coarse-to-Fine Video Retrieval Using Shot-Level Spatio-Temporal Statistics

Yu-Hsuan Ho, Chia-Wen Lin, *Senior Member, IEEE*, Jing-Fung Chen, and Hong-Yuan Mark Liao, *Senior Member, IEEE*

***Abstract*—In this paper, we propose a fast coarse-to-fine video retrieval scheme using shot-level spatio-temporal statistics. The scheme consists of a two-step coarse search followed by a fine search. In the coarse search stage, the shot-level motion and color distribution is computed as spatio-temporal features for shot matching. The first-step coarse search uses the shot-level global statistics to reduce the size of the search space drastically. By adding an adjacent shot of the first query shot, the second-step coarse search introduces a "causality" relation between two consecutive shots to improve the search accuracy. Finally, the fine-search step refines the search result by using the local color features extracted from the key frames of the query shots. Our experimental results show that the proposed method achieves good retrieval performance with a much reduced complexity compared to single-pass methods.**

***Index Terms*—Coarse-to-fine search, query by clip, video database, video matching, video retrieval.**

I. Introduction

With the ever-growing popularity of the Internet and the powerful computing capability of computers, efficient processing/retrieval of multimedia data has become an important issue. Multimedia is a general term that covers different media, including video, audio, graphics, text, images, or a combination of these media. Among the different types of media, video contains the most data and is relatively hard to deal with due to its complexity. To efficiently manage video data, including appropriate indexing, efficient storage and transmission, and fast retrieval, it is necessary to develop better video compression, indexing, and fast search algorithms. For fast transmission and efficient storage, MPEG video compression techniques are already well developed and widely deployed. However, the compressed video files still contain huge amounts of data, and the goal of efficient retrieval has yet to be realized. In the past decade, many crucial technologies, such as shot change detection [1], [2], shot representation [3], [4], key video frame/clip extraction [5], [6], and video sequence matching [7]–[10], have been developed to enhance video indexing and retrieval. The objective of the above-mentioned technologies is to reduce the amount of video data as much as possible, while simultaneously maintaining the information about the semantic content of the video. The basic unit of a video that holds semantics is the shot, which is a collection of successive video frames, and can be described simply as a continuous action in time or space [7]. After all shot boundaries have been detected, the following crucial issues must be addressed to enable fast retrieval: 1) how to group shots that are spatially close to each other and similar in content; 2) how to annotate a shot so that the subsequent retrieval task can be facilitated; and 3) how to design an algorithm to perform efficient video retrieval. In this paper, we propose a powerful video retrieval scheme to resolve these issues.

There can be little doubt that, in the multimedia era, using an unknown video clip to retrieve the complete counterpart video from a database will be an important future trend. Since a video shot is the basic semantic unit of a video clip, several shot-based approaches have been proposed for query-by-clip applications [10]–[14]. The method presented in [11] models the similarity between two shots as a graph diagram. It utilizes a color histogram and texture for shot matching and takes into account the duration of one shot when refining the matching results. In addition, a feedback scheme is used to improve the performance of the system. The method presented in [12] defines four kinds of criteria that are related to human perception. The four factors are combined with different weights to filter out dissimilar shots and speed up the comparison process. Meanwhile, the methods given in [10] and [13] model a video clip as a color feature trajectory. Although this is an efficient and simple feature representation, curve matching between two trajectories is computationally expensive. In [13], the authors use a preset match and tiling approach to retrieve similar video clips by finding the longest common subsequences (LCSs) within a sliding widow. In [14], on the other hand, the relation between two video clips is modeled as a graph. If the shots of two clips are similar, then an edge is assigned between them. Then, the clip that best matches a query is obtained by finding the graph with the most edges.

The video retrieval scheme we propose is a coarse-to-fine shot-based approach. Fig. 1 depicts a block diagram of our two-step coarse-search and one-step fine-search video retrieval scheme. The objective of the coarse search is to select a reasonably small number of candidate video clips from a video database, while avoiding nondetection of correct clips. At this stage, we compute the entropy of the motion vectors from every constituent shot of the query video clip and pick the shot with the maximum entropy within the clip as the query shot. The maximum entropy shot in a video clip usually has the most diverse object motions, presenting a relatively rare case in the clip. This makes it easier to find similar motion patterns than using shots with relatively uniform object motions, thereby providing more powerful discrimination ability for subsequently conducting an efficient search. The first step of the coarse search identifies a set of similar video clips by using shot-level spatio-temporal statistics (e.g., the motion and color histograms). This process significantly reduces the size of the search space. Then, in the second step, an adjacent (either

Manuscript received July 26, 2005; revised December 20, 2005. This paper was recommended by Associate Editor E. Izquierido.

Y.-H. Ho is with the Realtek Semiconductor Corporation, Hsinchu 300, Taiwan, R.O.C. (e-mail: yow@realtek.com.tw).

C.-W. Lin is with the Department of Computer Science and Information Engineering, National Chung Cheng University, Chiayi 621, Taiwan, R.O.C. (e-mail: cwlin@cs.ccu.edu.tw).

J.-F. Chen is with the Digital Media Center, National Taiwan Normal University, Taipei 106, Taiwan, R.O.C. (e-mail: jingfung@ntnu.edu.tw).

H.-Y. M. Liao is with the Institute of Information Science, Academia Sinica, Taipei 128, Taiwan, R.O.C. (e-mail: liao@iis.sinica.edu.tw).

Digital Object Identifier 10.1109/TCSVT.2006.873156

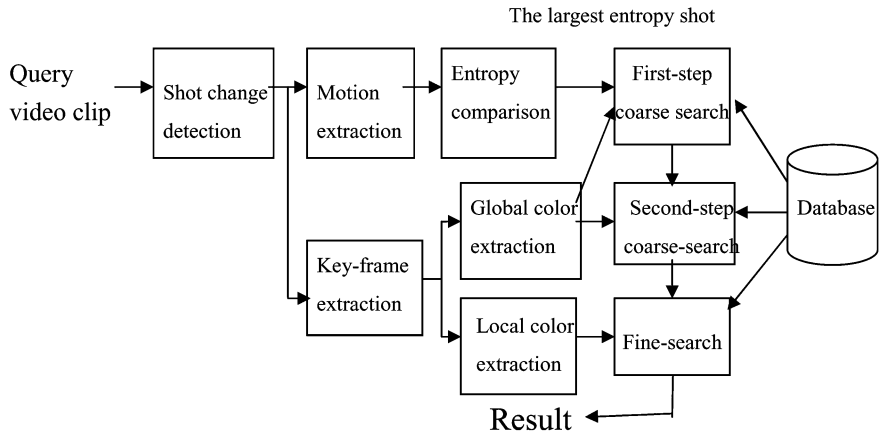

Fig. 1. Block diagram of our proposed multipass coarse-to-fine video retrieval scheme.

preceding or succeeding) shot to the first query shot is chosen, and the two shots are concatenated to form a two-shot query. In this step, the motion and color features used are the same as those used in the first step. However, a "causality" relation that defines the order of two consecutive shots is introduced to strengthen the discriminating capability. Because there usually exists in a video a temporal scenario (the "causality" relation) among consecutive video shots to constitute meaningful semantics, using two consecutive shots together for a search can take advantage of such a temporal scenario so as to retrieve more meaningful contents than using two nonconsecutive shots. In the coarse-search process, we extract the object motions of a query shot and quantize them into two-dimensional (2-D) probability distribution. The feature of this form is the temporal feature, which we use in our scheme. In addition, the color histogram of key frames extracted from the same shot using the method proposed in [6] is used as the spatial feature. To match two shots of different lengths, their corresponding motion and color probability distributions are compared using the discrete Bhattacharyya distance [15], which is specifically designed for comparing two arbitrary distributions. The joint distance, which sums up the distance of the motion statistics and that of the color histograms, is then used to measure the similarity between the two shots.

Following the two-step coarse search, a fine search is performed to enhance the retrieval accuracy. In the fine-search process, we extract color features from a set of selected key frames and use them to further refine the ranks of the matched video clips obtained in the coarse search. Each selected key frame is divided into four quarter-sized subimages, and the local color histogram of each subimage is calculated individually. We then calculate the Bhattacharyya distance and use it to choose the closest shots from the coarse-search outcomes.

The contribution of this study is twofold. First, we propose a new coarse-to-fine search paradigm which can significantly reduce computational complexity without sacrificing search accuracy. All of the search steps utilize the same motion and color features so that these features can be reused in each search step without further computation. Second, we introduce in the coarse-search step two new features: maximum motion entropy and causality of two consecutive shots, which provide higher level semantics to enhance the search accuracy with low extra cost.

The remainder of this paper is organized as follows. Section II describes the proposed two-step coarse-search scheme using the shot-level statistics and the causality between two consecutive query shots. Section III describes the fine-search scheme, which utilizes the local color distributions of key frames. Section IV contains the experimental results. Finally, in Section V, we present our conclusions.

II. Coarse Search Using Shot-Level Statistics and Causality

A. Preprocessing

Since a shot is the most primitive unit with semantic meaning that can be used for video retrieval, in our method, each video clip is segmented into its constituent video shots by using our previous method [2]. In order to extract local object motions as a feature for video retrieval, the global motion caused by camera operations must be excluded. We use the following four-parameter affine motion model to characterize the camera motion:

$$\mathbf{mv}_{\mathrm{cam}} = \begin{bmatrix} \text{zoom} & \text{rotate} \\ -\text{rotate} & \text{zoom} \end{bmatrix} \cdot \begin{bmatrix} mx_x \\ mx_y \end{bmatrix} + \begin{bmatrix} \text{pan} \\ \text{tilt} \end{bmatrix}. \quad (1)$$

We modify the compressed-domain global motion estimation method proposed in [16] to estimate $\mathbf{mv}_{\mathrm{cam}}$ between every two consecutive frames using the motion vectors of macroblocks carried in the compressed video. The input motion vectors are first filtered using a 2-D median filter with a 3×3 mask to remove the noise due to the inaccurate block-wise motion estimation performed in video encoding. The camera motion model is then obtained iteratively by minimizing the fitting error between the input motion vectors and the corresponding motion vectors generated from the estimated motion model using the Newton–Raphson method with outlier rejections [16]. The outlier rejections can improve the robustness of camera motion estimation by removing the unreliable motion vectors which tend to have largest fitting errors from the data set. On average, it takes three to five iterations to converge to a stable estimation.

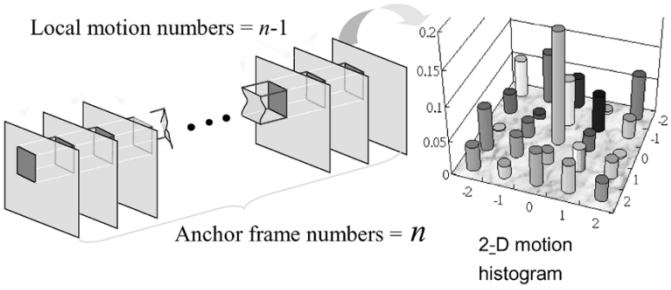

Fig. 2. Example showing how the motion vectors extracted from a macroblock sequence are projected onto the UV plane and calculated as a 2-D motion histogram.

Using the camera motions derived from the above model, the local motion corresponding to each macroblock sequence can be estimated by subtracting the input motion vectors from the estimated camera motions. The local motion vectors may have a wide dynamic range, leading to small populations for bins of motion vector values. Because it is not appropriate to compare the similarity of two motion distributions with bins of small populations, we transform the local motion vectors into a smaller domain. To reduce the dynamic range, a motion vector $(\mathrm{mv}_x, \mathrm{mv}_y)$ is transformed into the UV plane by the following quantization operation:

$$\mathbf{qmv} = \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} \mathrm{sgn}(\mathrm{mv}_x) \times \lfloor |\mathrm{mv}_x / I| + 0.5 \rfloor \\ \mathrm{sgn}(\mathrm{mv}_y) \times \lfloor |\mathrm{mv}_y / I| + 0.5 \rfloor \end{bmatrix} \quad (2)$$

where I is an integer used to control the degree of quantization and $\lfloor x \rfloor$ represents the largest integer smaller than x.

B. Proposed Two-Step Coarse Search

We now discuss how to calculate the statistics of motion from a valid macroblock sequence located in a shot. The left-hand side of Fig. 2 illustrates a typical shot consisting of n anchor frames. The following steps are performed to calculate the statistics of motion. First, let $\mathbf{m}_{i,j}$ represent the set of motion vectors of a valid macroblock sequence located in the ith row and jth column of the valid macroblock region. The probability that a quantized (or transformed) motion vector of this macroblock sequence falls into the bin (u, v) can be calculated as follows:

$$p(\mathbf{qmv} = (u,v) \,|\, \mathbf{qmv} \in \mathbf{m}_{i,j}) = \frac{\#\{\mathbf{qmv} \,|\, \mathbf{qmv} \in \mathbf{m}_{i,j}\}}{N_{\mathrm{MB}}} \quad (3)$$

where $\mathbf{qmv}$ represents a quantized motion vector obtained by (2) and N_{MB} is the total number of motion vectors in this valid macroblock sequence.

The right-hand side of Fig. 2 illustrates how to transform $n-1$ motion vectors into a normalized probability distribution map on the UV plane. In addition to the probability distribution calculated above, we also compute the entropy value of every valid macroblock sequence by the following equation:

$$H(S) = -\sum_{u,v} \{p(\mathbf{qmv} = (u,v) \,|\, \mathbf{qmv} \in \mathbf{m}_{i,j}) \times \ln p(\mathbf{qmv} = (u,v) \,|\, \mathbf{qmv} \in \mathbf{m}_{i,j})\}. \quad (4)$$

The calculation of the entropy shown in (4) is used to guide the selection of an unknown query clip from the constituent shots. As mentioned above, the shot with the largest entropy value usually contains the best discriminating information, which can be used to conduct an efficient search. In this study, a two-step process for efficient video retrieval is proposed. The complete shot with the largest entropy value mentioned above is adopted to execute the first-step coarse-search process. Then, in the second step, we adopt a shot that is adjacent to the chosen shot and concatenate them to achieve a more accurate search outcome. In other words, we consider the causality of two consecutive query shots.

After extracting the motion distributions of two distinct shots by (3), the matching process of the two shots is realized by comparing two probability distribution functions. Here, we use the discrete Bhattacharyya distance [15], given as follows, to perform the shot comparison task:

$$d(\mathbf{m}_{i,j}, \mathbf{m}'_{i,j}) = -\ln \sum_{u,v} \{p(\mathbf{qmv} = (u,v) \,|\, \mathbf{qmv} \in \mathbf{m}_{i,j}) \times p(\mathbf{qmv}' = (u,v) \,|\, \mathbf{qmv}' \in \mathbf{m}'_{i,j})\}^{1/2} \quad (5)$$

where $\mathbf{m}_{i,j}$ and $\mathbf{m}'_{i,j}$ represent the sets of quantized motion vectors extracted, respectively, from the valid macroblock sequence located at the (i,j)th locations of two distinct shots.

To calculate the overall Bhattacharyya distance between two arbitrary shots, we must accumulate the measured distances between all macroblock sequence pairs located in the valid macroblock region. The overall similarity $D(S, S')$ is calculated by

$$D(S, S') = \frac{\sum_{i,j} d(\mathbf{m}_{i,j}, \mathbf{m}'_{i,j})}{N} \quad (6)$$

where S and S' represent two distinct shots, and N represents the total number of valid macroblock sequences in a shot.

Besides motion, spatial color information is also an important feature in video retrieval. Color is a highly perceptive image feature that has been used extensively in previous research on content-based image retrieval (CBIR) [17]. The advantage of using color histograms is that scale invariance and a high degree of immunity to noise can be achieved. Because a color histogram is also a kind of probability distribution, the discrete Bhattacharyya distance can be used to measure the similarity between the color histograms of two shots as follows:

$$d(p,q) = -\ln \sum_i (p_{\mathrm{Y}}(i) \times q_{\mathrm{Y}}(i))^{1/2} - \ln \sum_i (p_{\mathrm{Cb}}(i) \times q_{\mathrm{Cb}}(i))^{1/2} - \ln \sum_i (p_{\mathrm{Cr}}(i) \times q_{\mathrm{Cr}}(i))^{1/2} \quad (7)$$

where p and q are two key frames of two arbitrary shots and $p_{\mathrm{Y}}(i), p_{\mathrm{Cb}}(i)$, and $p_{\mathrm{Cr}}(i)$ are the ith bins of the histogram. The subscripts Y, Cb, and Cr denote the three color components of the YCbCr format video.

Since a video has both spatial and temporal dimensions, video retrieval should capture the spatio-temporal content of video shots to obtain more accurate results. Therefore, we combine the similarity metrics of motion and color distribution in (5) and (7), respectively, for shot-level global feature matching to obtain the coarse-search results. To reduce computational complexity, the color distribution of a compressed video shot is extracted from the dc images of the shot. The dc image of each coded frame can be extracted directly from the compressed domain using the method described in [1]. The motion and color distributions of every video shot in the video database are offline extracted and stored in associated metadata files, whereas the distributions of input query shots are online extracted and compared with those in the metadata files.

In the first step of the coarse search, the shot with the largest entropy value in the query clip is utilized as the first query shot for video retrieval. This query returns a list of matched clips ranked in similarity order using the similarity metrics in (5) and (7). Then, in the second step, a shot adjacent to the first query shot is chosen, and the two consecutive shots are concatenated to form a two-shot query. This not only yields a more accurate result, but also reduces the size of the candidate list derived in the first step.

III. Fine Search Using Local Color Histogram of Key Frames

Because of its low computational requirement, the two-step coarse search procedure can efficiently select a relatively small set of candidate video clips from a large database. The coarse search results, however, may not be sufficiently accurate, since using only shot-level features may not be able to discriminate video shots with similar global statistics but with different spatial layouts and structures. Further extracting and storing the local spatial features for every frame of each shot, however, is not efficient in terms of computational complexity and storage costs. Since there is typically a high degree of similarity among the spatial features of neighboring frames, using the spatial features of the most representative key frames of a shot can usually achieve a retrieval accuracy level close to that of using all of the features of the shot, while reducing the computational complexity and storage cost drastically. In this study, the key frames of a video shot are extracted using our previous method presented in [6]. As a set of key frames have been chosen for each shot, the problem of video shot matching is how to determine the similarity between the key frames of two shots. This problem is similar to CBIR for individual key-frame matching.

After extracting the key frames, we divide each key frame into four quarter-sized subimages. Because we use the local spatial structures to further refine the search results, the color histogram of each subimage is calculated individually.

After calculating the local color histograms of each key frame, we combine the bins of all blocks to form a new distribution and use the color histogram distance in (7) to measure the similarity between the distributions of two shots to refine the results obtained from the coarse-search steps. By extracting key frames, a significant computational complexity saving can be achieved without sacrificing the retrieval performance. Table I compares the retrieval time of the fine search, with and without key frames, for 200 candidate video clips obtained in the coarse-search step when one key frame is extracted for one shot. The key-frame extraction takes only about 12% run-time of the feature extraction process. Compared with the whole search time, which involves feature extraction and shot matching, the complexity of key-frame extraction is not obvious.

TABLE I
Run-Time Comparison of Retrieval With Key Frames and With the Whole Shot

Retrieval / shot-length	with key frames	with whole shot
236	1 s	8 s
485	1 s	15 s

IV. Experimental Results

To demonstrate the effectiveness of the proposed method, we tested our algorithm on six digital videos comprising a total number of 1682 shots. The lengths of the six digital videos were: 55 min (503 shots, documentary, video #1); 52 minutes (405 shots, documentary, video #2); 29 minutes (241 shots, commercial, video #3); 38 minutes (193 shots, news, video #4); 38 minutes (283 shots, sports news, video #5); and 17 minutes (57 shots, home video, video #6). We chose these videos because of their variety.

In our experiments, 18 sample query video clips were selected from the 1682-shot database. For each sample clip, the ground truth of each relevant video clip was established by choosing a set of the most relevant video shots from the video database manually. In the proposed method, each sample query returns a list of matched clips ranked in similarity order. The retrieval performance is evaluated by the precision and recall rates.

A. Coarse-Search Performance

In the first step of the coarse search, a single shot with the maximum motion entropy is utilized to perform video retrieval. As mentioned, Fig. 3 shows the average precision-recall performance comparison of a motion-based coarse search between using maximum entropy shots and using randomly selected shots in a query clip for 18 queries. Obviously, using maximum motion entropy shots yields significantly better average precision-recall performance. After selecting the maximum entropy shot, we extract the shot's global motion and color statistics and compare them with the statistics prestored in the database. In the experiments, 200 candidate shots were retrieved during this first step using the shot-level motion and color statistics from the video database. That is, 100 candidate clips were obtained using the motion feature, and the other 100 clips were obtained using the color feature. The redundant results derived from both sets of statistics were then removed from the candidate list. Fig. 4(a) shows a sample query shot, and Fig. 4(b) shows the top three ranked results from the 1682-shot database, retrieved by the first-pass global search using global statistics matching.

Two concatenated shots are subsequently utilized to perform the second step of the coarse search, which reranks the candidate

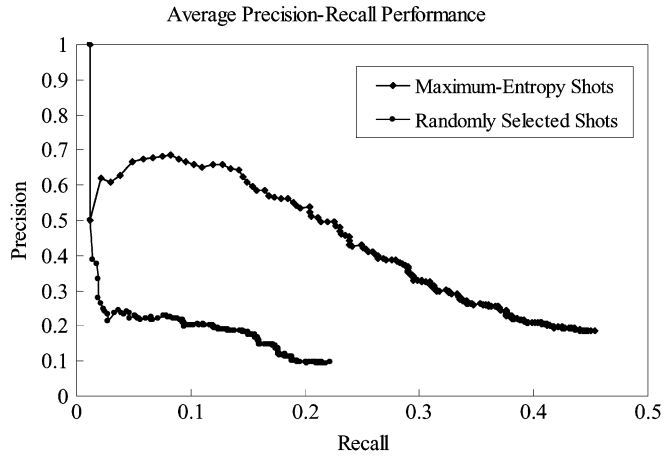

Fig. 3. Average precision-recall performance comparison between using maximum entropy shots and using randomly selected shots in a query clip for 18 queries.

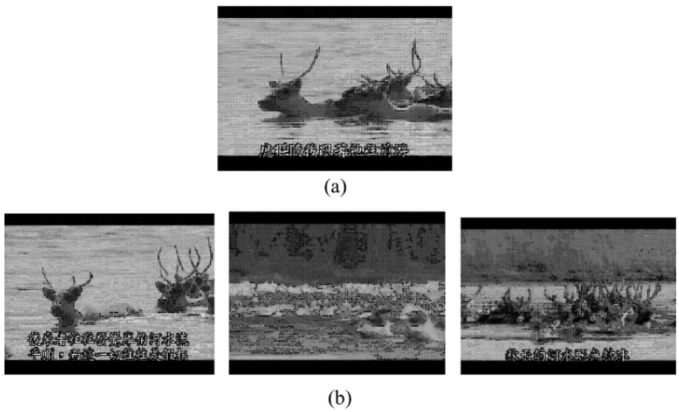

Fig. 4. Query example #1: (a) the query shot and (b) the top three shots retrieved from the 1682-shot database (the left shot is the top ranked one).

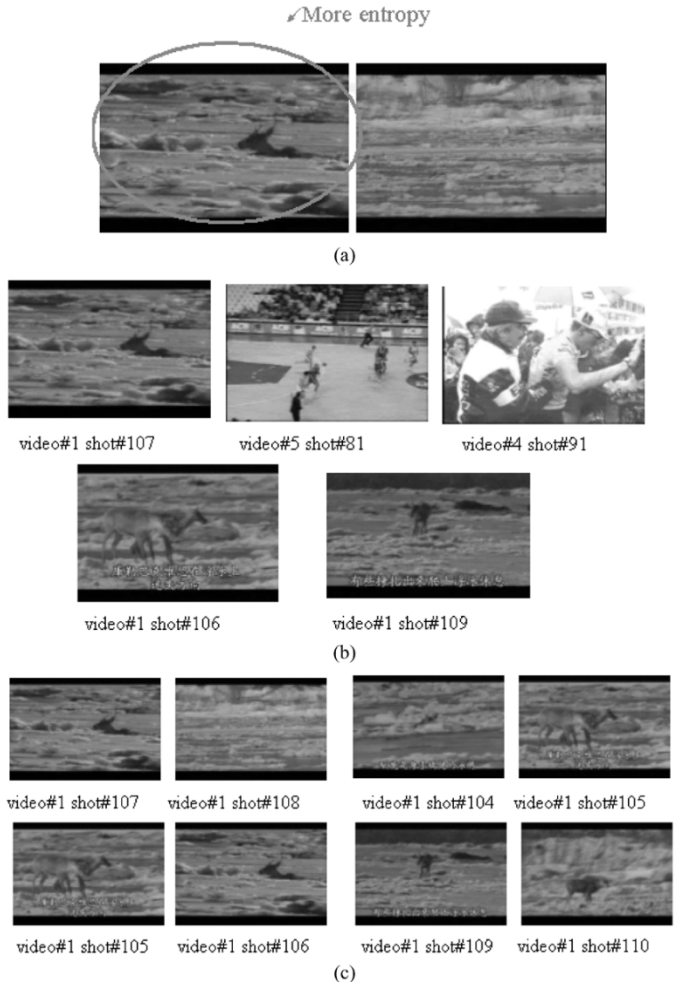

Fig. 5. Query example #2. (a) The query video clip with two shots. (b) The top five retrieved shots of the first pass from the 1682-shot database. (c) The top eight retrieved shots of the second step from the candidates provided by the coarse-search step.

list and reduces its size by 50% (i.e., 100 candidates in our experiments). Using the causality between two video shots effectively improves the coarse search result. For example, Fig. 5(a) and (b) shows that, although a coarse search based on global motion and color statistics is very efficient in terms of computational complexity, the search results may not be very accurate. In the second step of the coarse search, we add one more shot [the right-hand side of Fig. 5(a)] to the comparison process. Since the causality factor and the spatio-temporal statistics of one more shot are taken into account to enhance the power of the feature set, we observe that the top eight retrieved shots [Fig. 5(c)] are much more accurate with respect to the query shot pair.

B. Fine-Search Performance

To evaluate the performance of the proposed fine-search scheme, we chose a sports video clip comprised of two shots with multiple motion types for our test. Fig. 6(a) shows the first frame of the query video clip, Fig. 6(b) shows the top five retrieved results of the two-step coarse search, and Fig. 6(c) shows the top five retrieved results of the fine search using the local color histograms of the key frames. Clearly, the combined coarse-to-fine search yields more accurate results than the two-step coarse search alone, especially for video clips with multiple motion types. Table II lists the retrieved shots, the relevance of each retrieved shot to the query shot, and the precision and recall results of the two-step coarse search and the coarse-to-fine search, respectively, for a sample query video shot #107 of video #1. The tables show the retrieved video shots in decreasing order of similarity.

Fig. 7 shows the average precision-recall performance comparison of the 18 sample queries (three queries per test video) using the first step of the coarse search (using the shot-level motion statistics), the first two steps of the coarse search (with causality), and the coarse-to-fine search, respectively. The results show that the proposed coarse-to-fine search method achieves a better performance with significantly lower computational complexity when compared with single-pass methods.

V. Conclusion

In this study, we have presented a fast coarse-to-fine query-by-clip video retrieval scheme that consists of a two-step coarse search and a fine search. The first step of the coarse search utilizes the shot-level global statistics of motion and color of the shot with maximum entropy in the query clip as the

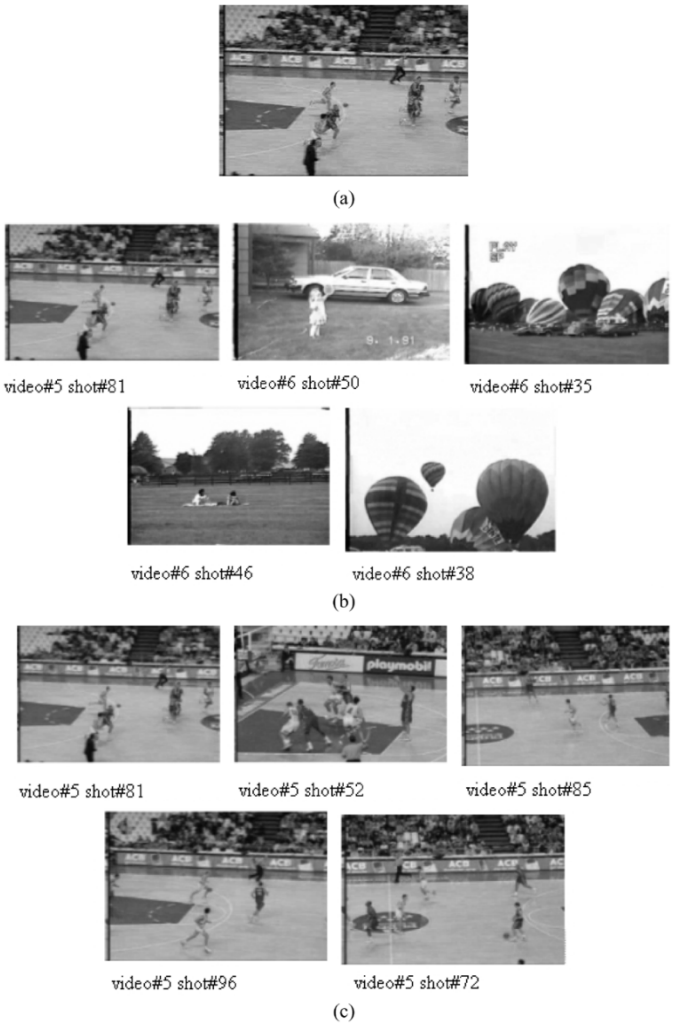

Fig. 6. Query example #3. (a) The query video shot. (b) The top five retrieved shots of the two-pass coarse search from the 1682-shot database. (c) The top five retrieved shots of the fine search using the local statistics of the key frames.

TABLE II
COMPARISON OF PRECISION AND RECALL VALUES FOR SHOT #107 OF VIDEO #1 USING TWO-STEP COARSE SEARCH (METHOD C) AND COARSE-TO-FINE SEARCH (METHOD C + F)

# of returned shots	Shot No.		Relevance		Precision		Recall	
	C	C+F	C	C+F	C	C+F	C	C+F
1	0_106	0_106	V	V	1.0	1.0	0.17	0.17
2	0_105	0_104	V	V	1.0	1.0	0.33	0.33
3	0_104	0_105	V	V	1.0	1.0	0.50	0.50
4	0_108	0_108	V	V	1.0	1.0	0.67	0.67
5	3_98	0_103	X	V	0.80	1.0	0.67	0.83
6	4_15	0_107	X	V	0.67	1.0	0.67	1.0
7	0_107	0_404	V	X	0.71	0.86	0.83	1.0
8	2_76	0_390	X	X	0.63	0.75	0.83	1.0
9	3_190	5_53	X	X	0.56	0.67	0.83	1.0
10	3_78	5_54	X	X	0.50	0.60	0.83	0.17

search features to quickly select a reasonably small set of candidate video shots from a video database. The causality between two consecutive shots is subsequently exploited in the second step of the coarse search to obtain a more accurate and smaller

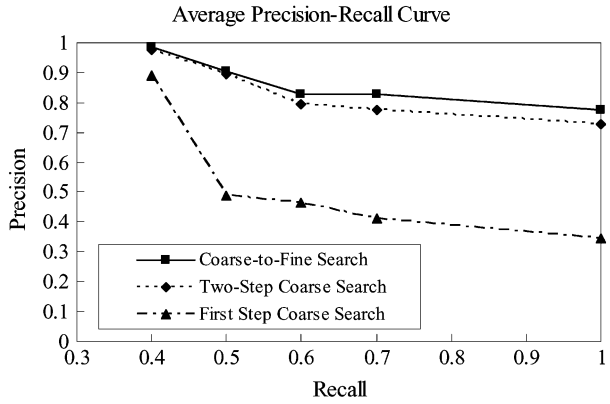

Fig. 7. Average precision-recall performance comparison of the first-step coarse search, the two-step coarse search, and the coarse-to-fine search for 18 queries.

set of search outcomes. The fine search then utilizes the local color histograms of key frames to determine the best matches from the set of candidate shots obtained in the coarse search. We have also proposed an efficient scheme for extracting the most representative key frames from a shot. Our experimental results show that the proposed method can provide satisfactory retrieval results with significantly lower complexity compared to traditional single-step schemes.

Although the proposed scheme can improve the efficacy of video retrieval, sometimes the low-level motion and color features used in the coarse and fine search steps may not be able to achieve accurate search results, since they lack high-level semantics. Incorporating the maximum motion entropy and causality features achieves performance improvement by introducing higher level semantics, but it may not fit a wide range of applications very well. We believe that introducing application-dependent high-level semantics (e.g., the motion flow presented in [18] for video surveillance applications) in the proposed coarse-to-fine paradigm would further enhance the search accuracy and efficiency. The major challenge would be how to extract useful high-level semantics from a compressed video with reasonable complexity.

REFERENCES

[1] B. Yeo and B. Liu, "Rapid scene analysis on compressed video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 5, no. 6, pp. 533–544, Dec. 1995.
[2] C.-W. Su, H.-Y. M. Liao, H.-R. Tyan, and L.-H. Chen, "A motion-tolerant dissolve detection algorithm," *IEEE Trans. Multimedia*, vol. 7, no. 6, pp. 1106–1113, Dec. 2005.
[3] H. S. Chang, S. Sull, and S.-U. Lee, "Efficient video indexing scheme for content-based retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 8, pp. 1269–1279, Dec. 1999.
[4] Y. F. Ma and H. J. Zhang, "A new perceived motion based shot content representation," in *Proc. IEEE Int. Conf. Image Process.*, Thessaloniki, Greece, Oct. 2001, vol. 3, pp. 7–10.
[5] H.-C. Lee and S.-D. Kim, "Iterative key frame selection in the rate-constraint environment," *Signal Process. Image Commun.*, vol. 18, pp. 1–15, 2003.
[6] Y.-H. Ho, W.-R. Chen, and C.-W. Lin, "A rate-constrained key-frame extraction scheme for channel-aware video streaming," in *Proc. IEEE Int. Conf. Image Process.*, Singapore, Oct. 2004, vol. 1, pp. 613–616.
[7] M. M. Yeung and B. Liu, "Efficient matching and clustering of video shots," in *Proc. IEEE Int. Conf. Image Process.*, Washington, DC, USA, Oct. 1995, vol. 1, pp. 338–341.

[8] S. H. Kim and R. H. Park, "An efficient algorithm for video sequence matching using the modified Hausdorff distance and the directed divergence," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 12, no. 7, pp. 592–596, Jul. 2002.

[9] S. Cheung and A. Zakhor, "Efficient video similarity measurement with video signature," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 1, pp. 59–74, Jan. 2003.

[10] M. R. Naphade, M. M. Yeung, and B. L. Yeo, "A novel scheme for fast and efficient video sequence matching using compact signatures," in *Proc. SPIE Conf. Storage Retrieval Media Databases*, Jan. 2000, vol. 3972, pp. 564–572.

[11] Y. Wu, Y. Zhuang, and Y. Pan, "Content-based video similarity model," in *Proc. ACM Int. Conf. Multimedia*, Marina del Rey, CA, Oct. 2000, pp. 465–467.

[12] X. Liu, Y. Zhuang, and Y. Pan, "A new approach to retrieve video by example video clip," in *Proc. ACM Int. Conf. Multimedia*, Orlando, FL, Oct. 1999, pp. 41–44.

[13] L. Chen and T.-S. Chua, "A match and tiling approach to content-based image retrieval," in *Proc. IEEE Int. Conf. Multimedia Expo*, Tokyo, Japan, Aug. 2001, pp. 301–304.

[14] Y.-X. Peng, C.-W. Ngo, Q.-J. Dong, Z.-M. Guo, and J.-G. Xiao, "Video clip retrieval by maximal matching and optimal matching in graph theory," in *Proc. IEEE Int. Conf. Multimedia Expo*, Baltimore, MD, Jul. 2003, pp. 317–320.

[15] L.-F. Chen, H.-Y. M. Liao, J.-C. Lin, and C.-C. Han, "Why recognition in a statistics-based face recognition system should be based on the pure face portion: A probabilistic decision-based proof," *Pattern Recognit.*, vol. 34, no. 5, pp. 1393–1403, 2001.

[16] Y. Su, M.-T. Sun, and V. Hsu, "Global motion estimation from coarsely sampled motion vector field and the applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 2, pp. 232–242, Feb. 2005.

[17] A. D. Bimbo, *Visual Information Retrieval*. San Mateo, CA: Morgan Kaufmann, 1999.

[18] C.-W. Su, H.-Y. M. Liao, K.-C. Fan, C.-W. Lin, and H.-R. Tyan, "A motion-flow-based fast video retrieval system," in *Proc. ACM SIGMM Int. Workshop Multimedia Inf. Retrieval*, Singapore, Nov. 2005, pp. 105–112.