5-2006

Clip-based similarity measure for query-dependent clip retrieval and video summarization

Yuxin PENG

Chong-wah NGO
Singapore Management University, cwngo@smu.edu.sg

Citation

1

Clip-Based Similarity Measure for Query-Dependent Clip Retrieval and Video Summarization

Yuxin Peng and Chong-Wah Ngo, *Member, IEEE*

***Abstract*—This paper proposes a new approach and algorithm for the similarity measure of video clips. The similarity is mainly based on two bipartite graph matching algorithms: maximum matching (MM) and optimal matching (OM). MM is able to rapidly filter irrelevant video clips, while OM is capable of ranking the similarity of clips according to visual and granularity factors. We apply the similarity measure for two tasks: retrieval and summarization. In video retrieval, a hierarchical retrieval framework is constructed based on MM and OM. The validity of the framework is theoretically proved and empirically verified on a video database of 21 h. A query-dependent clip segmentation algorithm is also proposed to automatically locate the potential boundaries of clips in videos. In video summarization, a graph-based clustering algorithm, incorporated with the proposed similarity measure, is adopted to detect the highlighted events reported by different newscasts.**

***Index Terms*—Clip similarity, query-based segmentation, hierarchical video retrieval, summarization.**

I. Introduction

DUE TO THE drastic advances in multimedia and Internet applications, more effective techniques for video retrieval and summarization are in increasing demand. One critical component in these techniques is the similarity measure of visual information. While the issues in shot-based similarity have been intensively addressed for retrieval, clustering, and summarization, clip-based similarity remains a difficult problem that has not yet been fully exploited. In this paper, we propose a hierarchical framework based on the bipartite graph matching algorithms for the similarity filtering and ranking of video clips.

A shot is a series of frames with continuous camera motion, while a clip is a series of shots that are coherent from the narrative as well as the users point of view. Depending on the types of video genre being considered, a clip is normally referred to as "scene" [2], [3], "story" [10], [14], [27], and "news story" [8]. In [2] and [3], a scene is defined as a collection of consecutive shots which are related by some spatial context. The context can be an event, activity, or a physical location. In [14], a story is a single dramatic event that comprises shots taken by a small number of related cameras, while, in [10], story (or, more precisely, logical story unit) is an approximation of a movie episode. In [8], a news story is defined as a segment of news broadcast with a coherent news focus containing at least two independent and declarative clauses. In this paper, we do not attach any specific constraint to the definition of clip. The exact definition depends on the type of a given query. Clip can be a superset of scene, story, and news story, where the underlying shots are captured in one or different physical locations by various (related and unrelated) cameras. A clip, like scene, story, and news story, usually conveys one meaningful event. Basically, shot-based retrieval is useful for tasks such as the detection of known objects and certain kinds of videos like sports. For most general videos, 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 the full story. For most casual users, query-by-clip is definitely more concise and convenient than query-by-shot.

Existing approaches in clip-based retrieval include [1], [5], [6], [7], [9], [11]–[13], [15], [16], [18], [24], and [25]. Some research has focused on the rapid identification of similar clips [6], [7], [11], [13], [18], while others emphasize the similarity ranking of videos clips [1], [5], [9], [12], [15], [16], [24], [25]. In [6], [7], and [13], fast algorithms are proposed by deriving signatures to represent the clip contents. The signatures are basically the summaries or global statistics of low-level features in clips. The similarity of clips depends on the distance between signatures. Global signatures are suitable for matching clips with almost identical content but little changes due to compression, formatting, or minor editing in spatial or temporal domain. One successful example is the high accuracy and speed in retrieving commercial clips from large video databases [13].

The similarity ranking of clips, in general, is built upon the shot-based retrieval. Besides relying on shot similarity, clip similarity is also dependent on the interrelationship such as temporal order, granularity, and interference among shots. Granularity models the degree of one-to-one shot matching between two clips, while interference models the percentage of unmatched shots. Recent approaches in clip similarity ranking include [1], [5], [9], [12], [15], [16], [24], and [25]. A brief summary of these approaches is given in Table I. In [5], a window is slid across the video to investigate the matched shots. The number of matched shots in a window is put at the center position of the window. This forms a one-dimensional

Manuscript received July 19, 2004; revised March 10, 2005 and September 9, 2005. This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China, under Grant CityU 1072/02E and the City University of Hong Kong under Project 7001546. This paper was recommended by Associate Editor R. Lienhart.

Y. Peng is with the Institute of Computer Science and Technology, Peking University, Beijing 100871, China, and the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: pengyuxin@icst.pku.edu.cn).

C. W. Ngo is with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong (e-mail: cwngo@cs.cityu.edu.hk).

Digital Object Identifier 10.1109/TCSVT.2006.873157

TABLE I
COMPARISON OF SOME APPROACHES IN CLIP-BASED SIMILARITY RANKING

	Sliding window [5]	Temporal order [9], [12], [24]	*V*string edit [1]	Bi-directional matching [12]	Cluster-based [16], [25]	Multi-level [15]
Shot matching	match & tile	order preserving matching	video specific edit operations	two most similar shots	pairwise similarity clustering	sequence or set matching
Temporal order	×	√	√	×	√	√
Granularity	×	√	√	×	√	Partially
Interference	×	×	√	×	√	×
Clip filtering	×	×	×	×	×	×
Online clip segmentation	√	×	×	×	×	×

(1-D) curve with the x-axis as a shot sequence and the y-axis as the number of matches. The relevant clips are then segmented by locating the local maxima of curve. The major disadvantage of [5] is that the granularity, temporal order, and interference are not taken into account. One typical example is that the similarity of two clips with one-to-one shot matching can be the same as two clips with one-to-many matching.

In [9], [12], and [24], shots in two clips are matched by preserving their temporal order. For instance, [24] employs dynamic programming to align two sequences in time order and measures the similarity accordingly. In clip retrieval, shot matching by time preserving, however, may not be appropriate, since shots in different clips tend to appear in various orders due to editing effects. Even for a commercial video, several editions are normally available with various shot orders and durations. In [12], a nontemporal preserving matching is also proposed. The similarity of clips is reduced to the similarity of two most similar shots by bidirectional matching. Nevertheless, this approach is prone to noise and, furthermore, the similarity ranking is ineffective since two clips can be considered similar even if only one pair of shots is matched.

Several sophisticated approaches for clip retrieval are proposed in [1], [16], and [25], where different factors including granularity, order, and interference are taken into account. In [1], several new edit operations (swap, fusion and break) are proposed along with the parametric edit distances which consider the weights of traditional (i.e., insert, delete, and substitute) and video-specific operations for similarity ranking. In [16] and [25], a cluster-based algorithm with pairwise similarity among shots as input is employed to match similar shots. The aim of clustering is to find a cut (or threshold) that can maximize the centroid distance of similar and dissimilar shots. The cut value is used to decide whether two shots should be matched. In [15], a multilevel matching scheme is proposed to recursively measure clip similarity at shot-scene-video levels. At the shot level, two types of representation (sequence and set) are proposed for matching. Similar to [16] and [25], empirical thresholds are set to find the set of matching shots. At the video level, two measures, resequence and correspondence, are used to assess the similarity of clips. The correspondence measure can partially evaluate the degree of granularity.

As indicated in Table I, most approaches [6], [7], [9], [12], [15], [16], [24], [25] assume that video clips are presegmented and always available for matching. In addition, the capabilities of filtering irrelevant clips prior to similarity ranking are usually not considered [5]–[7], [9], [11], [12], [15], [16], [24], [25]. Our proposed similarity measure is in line with [16] and [25], but with the capabilities of clip filtering and online segmentation. Instead of adopting a cluster-based algorithm as in [16] and [25], we formulate the problem of shot matching as a bipartite graph matching in two stages. In the first stage, the candidate clips are located and segmented from videos while the irrelevant clips are rapidly filtered. In the second stage, the detailed similarity ranking is conducted by considering the quality of matching determined jointly by the granularity, temporal order, and interference factors. The major contributions of our approach are as follows.

- *Matching and filtering*. We adopt two bipartite graph matching algorithms, namely maximum matching (MM) and optimal matching (OM), for the matching of shots in clips. Both algorithms are constrained under one-to-one mapping. MM, by computing the maximum cardinality of matching, is capable of rapidly filtering irrelevant clips. The goal of clip filtering is to rapidly prune irrelevant clips and thus improve the speed efficiency of retrieval. With MM, there are less video clips to be considered by OM. OM, by optimizing the total weight of matching, is able to rank relevant clips based on the similarity of visual and granularity. MM and OM can thus form a hierarchical framework for filtering and retrieval. By the definitions of MM and OM [17], [26], the validity of the hierarchical framework can be justified by showing that MM never filters clips that are considered to be similar by OM.
- *Similarity ranking*. The clip similarity is jointly determined by visual, granularity, order, and interference factors. While visual and granularity are measured by OM, temporal order similarity is evaluated effectively by dynamic programming. The measure of interference is based on the output of OM.
- *Query-dependent clip segmentation*. The segmentation of videos into clips is implicitly tailored to the content of a query clip. Given a query and a video, a bipartite graph is constructed by many-to-many mapping. The mapping usually results in the following properties: some shots in the video are densely matched along the temporal dimension, while most shots are sparsely matched or unmatched. Our

algorithm automatically locates and segments the dense regions as potential candidate clips. The boundaries of segmented clips can be fine-tuned by OM.

Besides applying the proposed similarity measure for video retrieval, we also demonstrate the effectiveness of clip similarity for video summarization. Given 10 h of videos collected from different newscasts, we adopt a graph-based clustering approach, incorporated with the proposed clip similarity, to automatically detect highlighted events and then generate summaries of different time lengths.

The remainder of this paper is organized as follows. Section II describes the preprocessing steps including shot boundary detection and shot similarity measure. Section III presents the proposed clip-based similarity measurement by MM and OM. Section IV justifies the validity of the hierarchical video retrieval framework formed by MM and OM. The algorithm for online video clip segmentation is also presented. Section V describes the application of clip similarity for video summarization. Section VI shows experimental results, while Section VII concludes this paper.

II. Video Preprocessing

Preprocessing includes shot boundary detection, key-frame representation, and shot similarity measure. All processing is carried out in the compressed video domain. The information of shots and key frames are stored and indexed in a database for clip-based retrieval. We adopt the detector in [19] for the partitioning of videos into shots. The detector can locate as well as classify cuts, wipes, and dissolves. Motion-based analysis in [20] is then employed for compact video representation. Basically, key frames are selected and constructed adaptively from shots based on motion content. For instance, a sequence with pan is represented by a panoramic key frame (mosaic), while a sequence with zoom is represented by two frames before and after the zoom. This scheme is also similar to the compact shot representation in [2] and [3] for scene representation and clustering.

Let the key frames of a shot s_i be $\{r_{i1}, r_{i2}, \ldots\}$ and the similarity between two shots is defined as

$$\mathrm{Sim}(s_i, s_j) = \frac{1}{2}\{\phi(s_i, s_j) + \hat{\phi}(s_i, s_j)\} \tag{1}$$

where

$$\phi(s_i, s_j) = \max_{p=\{1,2,\ldots\}, q=\{1,2,\ldots\}} \mathrm{Intersect}(r_{ip}, r_{jq})$$
$$\hat{\phi}(s_i, s_j) = \widehat{\max_{p=\{1,2,\ldots\}, q=\{1,2,\ldots\}}} \mathrm{Intersect}(r_{ip}, r_{jq}).$$

The similarity function $\mathrm{Intersect}(r_{ip}, r_{jq})$ is the color histogram intersection[1] of two key frames r_{ip} and r_{jq}. The function $\widehat{\max}$ returns the second largest similarity value among all pairs of key-frame comparisons. The disadvantage of using color histograms is that two key frames will be considered similar as long as they have similar color distribution, even though their contents are different. To increase robustness, we require at least two pairs of key frames with similar color histograms. Notice that ϕ and $\hat{\phi}$ cannot remedy the lack of spatial information in histogram comparison. Indeed, $\hat{\phi}$ is used to moderate the similarity value ϕ. The histogram is represented in hue, saturation, and value (HSV) color space. Hue is quantized into 18 bins, while saturation and value are quantized into three bins respectively. The quantization provides 162 $(18 \times 3 \times 3)$ distinct color sets. In our approach, the content of a shot is characterized based on motion as in [20]. A short shot with different camera motions, for instance, can have more key frames than a long shot with no motion or single camera motion. Therefore, the similarity measurement is not biased when comparing a very short shot with similar color content to a long shot, although shot length is not taken into account.

III. Clip-Based Similarity Measure

The similarity is based mainly on MM and OM. Both MM and OM are classical matching algorithms in graph theory [17], [26]. MM computes the maximum cardinality matching in an unweighted bipartite graph, while OM optimizes the maximum weight matching in a weighted bipartite graph.

A. Notation

For the ease of understanding, we use the following notations in the remainder of this paper.

- Let $X = \{x_1, x_2, \ldots, x_p\}$ be a query clip with p shots and x_i represents a shot in X.
- Let $Y_k = \{y_1, y_2, \ldots, y_q\}$ be the kth video clip with q shots in a video $\mathbf{Y}$ and y_j is a shot in Y_k.
- Let $G_k = \{X, Y_k, E_k\}$ represents a bipartite graph constructed by X and Y_k. $V_k = X \cup Y_k$ is the vertex set while $E_k = \{\omega_{ij}\}$ is the edge set. For an unweighted graph, $\omega_{ij} = \{0, 1\}$ and 1 represents that there is an edge (or a match) from shot x_i to shot y_j. For a weighted graph, ω_{ij} represents the shot similarity between x_i and y_j.

B. Definition

Given two set of vertices, MM aims to maximize the cardinality of matching, and OM aims to optimize the total weights of matching. The following definitions rigorously explain the terms "matching," "maximum cardinality matching," and "optimal matching" in a bipartite graph. These definitions are further used in Section IV-A for the verification of hierarchical retrieval framework.

Definition 1: Denote a bipartite graph as $G_k = \{X, Y_k, E_k\}$; $M \subseteq E$, is a match if any two edges in M are not adjacent.

Definition 2: Suppose that $\hat{M}$ contains the matched pairs in G_k and satisfies Definition 1. Then, $\hat{M}$ is the maximum cardinality matching if there exists no matching M in G_k such that $|M| > |\hat{M}|$.

Definition 3: Let $\hat{M}$ contain the matched pairs in G_k and satisfy Definition 1. Then, $\hat{M}$ is the optimal matching in G_k if

$$\Omega(\hat{M}) = \max\{\Omega(M) | M \text{ is a matching in } G_k\} \tag{2}$$

where

$$\Omega(M) = \sum_{\omega_{ij} \in M} \omega_{ij} \tag{3}$$

[1] Histogram normalization is a problem when intersecting key frames of different sizes. We use the scheme in [20] where the result of intersection is normalized by the size of the smaller image.

is the sum of similarity in M and ω_{ij} is the similarity between shot i and shot j.

C. Video Clip Filtering by MM

Given X and Y_k, an unweighted bipartite graph G_k is formed by

$$\omega_{ij} = \begin{cases} 1, & \text{Sim}(x_i, y_j) > \mathcal{T} \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

The function Sim is based on (1). A threshold[2] $\mathcal{T}$ is set to determine whether there is an edge from x_i to y_j. Since a clip is composed of a series of shots with the same semantic, the color content of shots are usually intercorrelated and similar. Due to this self-similarity property, one shot in X can usually match multiple shots in Y_k. As a consequence, the mapping of shots in G_k is usually the many-to-many relationship. To maximize the matching of shots between X and Y_k under the one-to-one mapping constraint, MM is used due to its effectiveness and efficiency. The output of MM is a bipartite graph G_{MM} with each x_i matches with at most one y_j, and vice versa. Based on the number of edges in G_{MM}, we can rapidly filter dissimilar video clips while retaining only potentially relevant clips for the detailed similarity ranking. In general, if only a few shots in X can match Y_k, then Y_k should be considered as dissimilar to the query clip X. In our case, we define two clips as dissimilar if $|M| < (|X|/2)$, where $|M|$ is the number of edges in G_{MM} and $|X| = p$ is the number of shots in a query clip.

We employ the maximum cardinality matching algorithm (Kuhn algorithm) for the implementation of MM [26]. The details are given in Algorithm 1. The computational complexity of MM is $O(nm)$, where $n = p + q$ is the number of vertices (shots) and m is the number of edges in G_k.

Algorithm 1: Algorithm for Maximum Matching

```
1. $M \leftarrow \emptyset$
2. If all the vertices in $X$ have been tested, $M$ is the maximum
       matching of $G_k$ and the algorithm ends. Otherwise, go to step 3.
3. Find a vertex $x_i \in X$ that has not been tested. Set $A \leftarrow \{x_i\}$
       and $B \leftarrow \emptyset$.
4. Let $N(A) \subseteq Y_k$ be the set of vertices that match the vertices
       in set $A$. If $N(A) = B$, $x_i$ cannot be assigned to $M$. Label
       $x_i$ as tested, and then go to step 2. Otherwise, go to step 5.
5. Randomly find a vertex $y_j \in N(A) - B$.
6. If there exists a node $z \in X$ such that $(z, y_j) \in M$, set A
       $\leftarrow A \cup \{z\}$, $B \leftarrow B \cup \{y_j\}$and go to step 4. Otherwise,
       go to step 7.
7. Given $y_j$, find $z$ adjoined to $y_j$. This can be done as
       $y_i \in N(A)$.
8. Given $z$, find $y' \in B$ such that $(z, y') \in M$. This can be done
       because $A$ and $B$ contain only the end points of edges in $M$.
9. Repeat steps 7 and 8 until the initial vertex $x_i$ is found in
       step 7. The result is a path $P$ from $y_i$ to $x_i$. Such a path is
       called an augmenting path of $M$, defined by the property
       that neither end points of $P$ are part of $M$ but every other
       edge in $P$ belongs to $M$. Let $E(P)$ as the edge set of $P$.
10. Label $x_i$ as tested and set $M \leftarrow M \oplus E(P)$, where
       $M \leftarrow M \oplus E(P)$ is a sub-graph that includes all the edges
       in $M$ or $E(P)$ but not both.
```

D. Video Clip Ranking

The similarity ranking is based on the visual, granularity, temporal, and interference factors. These factors are mainly based on the results of OM which enforces the one-to-one mapping between two clips. While the visual and granularity factors are computed directly based on the output of OM, the temporal and interference factors are measured, respectively, by dynamic programming and the number of matched shots.

E. OM

Based on a weighted bipartite graph G_k formed by applying $\mathcal{T}$ as in (4), OM is employed to maximize the total weight of matching under the one-to-one mapping constraint. The output of OM is a weighted bipartite graph G_{OM} where one shot in X can match with at most one shot in Y_k and vice versa. The similarity of X and Y_k is assessed based on the total weight in G_{OM} as follows:

$$\text{Sim}_{\text{OM}}(X, Y_k) = \frac{\sum \text{Sim}(x_i, y_j)}{p} \quad \text{and} \quad \text{Sim}(x_i, y_j) > \mathcal{T} \tag{5}$$

where the similarity is normalized by the number of shots p in the query clip X. The implementation of OM is based on the Kuhn–Munkres algorithm [26]. The details are given in Algorithm 2. The running time of OM is $O(n^4)$, where $n = p + q$ is the total number of vertices in G_k.

Algorithm 2: Algorithm for Optimal Matching

```
1. Start with the initial label of $l(x_i) = \max_j\{\omega_{ij}\}$
       and $l(y_j) = 0$, where $i = 1, 2, \ldots, p$ and $j = 1, 2, \ldots, q$.
2. Compute $E_l = \{(x_i, y_j) | l(x_i) + l(y_j) = \omega_{ij}\}$,
       $G_l = (X, Y_k, E_l)$ and one matching $M$ in $G_l$.
3. If $M$ contains all the vertices in $X, M$ is the optimal
   matching
       of $G_k$ and the algorithm ends. Otherwise, go to step 4.
4. Find a vertex $x_i \in X$ that is not inside $M$. Set $A \leftarrow \{x_i\}$
       and $B \leftarrow \emptyset$.
5. Let $N_{G_l}(A) \subseteq Y_k$ be the set of vertices that matches the
       vertices in set $A$. If $N_{G_l}(A) = B$, then go to step 9.
       Otherwise, go to step 6.
```

[2] To ensure a high recall rate, the value of $\mathcal{T}$ is set as low as possible. $\mathcal{T}$ will not be sensitive to the final matching since the edges of dissimilar shots will not meet the one-to-one constraint in MM and will ultimately be filtered.

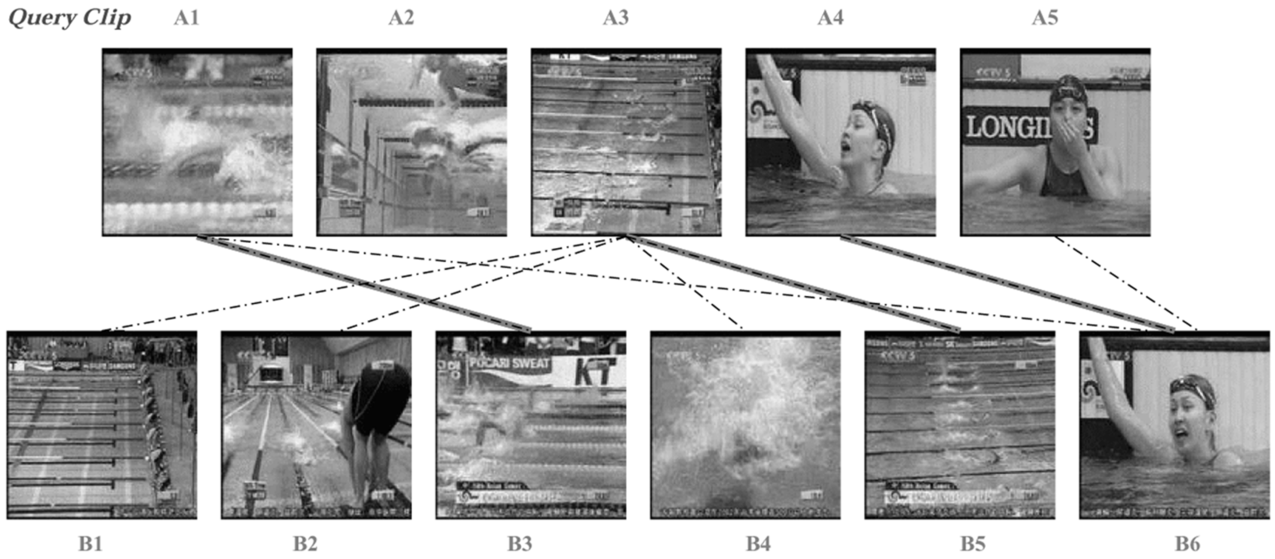


Fig. 1. Shot matching in two clips (dotted line: initial match; solid line: final match).

6. Randomly find a vertex $y_j \in N_{G_l}(A) - B$.

7. If there exists a node $z \in X$ such that $(z, y_j) \in M$, set $A \leftarrow A \cup \{z\}$, $B \leftarrow B \cup \{y_j\}$ and go to step 5. Otherwise, go to step 8.

8. Similar to MM, we can find an augmenting path P from x_i to y_j. Set $M \leftarrow M \oplus E(P)$ and go to step 3.

9. Compute $a = \min_{x_i \in A, y_j \notin N_{G_l}(A)} \{l(x_i) + l(y_j) - \omega_{ij}\}$, then construct a new label $\hat{l}(v)$ by

$$\hat{l}(v) = \begin{cases} l(v) - a & v \in A \\ l(v) + a & v \in B \\ l(v) & \text{otherwise} \end{cases}$$

Compute $E_{\hat{l}}$, $G_{\hat{l}}$ based on $\hat{l}$.

10. Set $l \leftarrow \hat{l}$, $G_l \leftarrow G_{\hat{l}}$, go to step 5.

F. Dynamic Programming (DP)

Given a bipartite graph G_{OM} computed by OM, the similarity of two clips based on the temporal order of shot matching can be formulated by DP. Denote $\mathcal{C}$ as a cost matrix indicating the number of shot pairs that are matched along the temporal order, we have (6), shown at the bottom of the page, where M is optimal matching that contains the set of matched pairs formed by OM. The running time of (6) is $O(pq)$, where p and q are, respectively, the number of shots in X and Y_k. The similarity between two clips based on the temporal order is defined as

$$\text{Sim}_{\text{DP}}(X, Y_k) = \frac{\mathcal{C}[p, q]}{p}. \tag{7}$$

G. Interference Factor (IF)

The IF counts the number of unmatched shots in $G_{\text{OM}}, i.e., p + q - 2 \times |M|$. The similarity between two clips based on IF is

$$\text{Sim}_{\text{IF}}(X, Y_k) = \frac{2 \times |M|}{p + q}. \tag{8}$$

Since the values of $|M|$, p and q are known, $\text{Sim}_{\text{IF}}(X, Y_k)$ can be computed in $O(1)$ time.

H. Clip Similarity

Given X and Y_k, the similarity is measured jointly by the degree of granularity and visual similarity, the temporal rder of matching, and interference factor as follows:

$$\text{Sim}_{\text{clip}}(X, Y_k) = \sum_{i \in \{\text{OM}, \text{DP}, \text{IF}\}} \alpha_i \text{Sim}_i(X, Y_k) \tag{9}$$

where $\sum_i \alpha_i = 1$ are the weights of different similarity measures. The value of α_i controls the ranking of similar video clips. In most video retrieval related tasks, the degree of granularity and visual similarity, which reflect the number and proximity of matching shots, respectively, should carry more weight than temporal order and interference factor. Thus, we set $\alpha_{\text{OM}} > \alpha_{\text{DP}} = \alpha_{\text{IF}}$ ($\alpha_{\text{OM}} = 0.4$, $\alpha_{\text{DP}} = \alpha_{\text{IF}} = 0.3$) in our experiments. These values can also be set based on user preference. Fig. 1 illustrates an example for clip similarity measure. The initial matches form a bipartite graph G_k with many-to-many mapping. For instance, shots A1, A4, and A5 are matched with B6, while shot A3 is matched with B1, B2, B4, and B5. By applying OM, three pairs of shots are optimally matched. All matched

$$\mathcal{C}[i, j] = \begin{cases} 0, & i = 0 \text{ or } j = 0 \\ \mathcal{C}[i-1, j-1] + 1, & i, j > 0, \quad (x_i, y_j) \in M \\ \max\{\mathcal{C}[i, j-1], \mathcal{C}[i-1, j]\}, & i, j > 0, \quad (x_i, y_j) \notin M \end{cases} \tag{6}$$

pairs are in order and five shots remain unmatched. Assuming that each pair has a similarity of 0.8, the visual and granularity factor is $(3 \times 0.8)/(5)$. The similarities based on temporal order and inference are $(3)/(5)$ and $(6)/(11)$, respectively.

IV. Video Retrieval

The retrieval of video clips can be conducted by the similarity measure based on OM, DP, and IF. However, since the total complexity is $O((p+q)^4) + O(pq) + O(1)$ for each comparison, the algorithm is inefficient, particularly in a large video database. The properties of MM and OM, nevertheless, allow us to effectively set up a hierarchical framework for efficient retrieval. The complexity of MM is $O(pq) + O((p+q)m)$, where $O(pq)$ is for similarity computation and $m = pq$ in the worst case. Notice that m is usually smaller than pq after applying (4). MM can be employed to efficiently filter irrelevant video clips. The combination of OM, DP, and IF has a higher time complexity but is more effective in similarity measure. They can serve to rank only those clips retained by MM.

A. Hierarchical Framework

To construct the hierarchical framework, we need to show that MM does not filter any video clip that is not also filtered by OM as well. In other words, if $\mathcal{R}_1$ and $\mathcal{R}_2$ are sets of similar clips retained by MM and OM, respectively, then $\mathcal{R}_2 \subseteq \mathcal{R}_1$. If the claim is correct, the hierarchical framework is not only efficient but also as effective as using OM, DP, and IF alone. The claim can be proved based on the definition of MM and OM in graph theory as follows.

Theorem 1:

Let $|\mathrm{MM}|$ be the number of edges by MM and $|\mathrm{OM}|$ be the number of edges by OM, under the same bipartite graph setting as (4) (i.e., the edges with similarity below or equal to $\mathcal{T}$ are removed from consideration). Then

$$|\mathrm{MM}| \geq |\mathrm{OM}|. \tag{10}$$

Proof: By (2) in Definition 3, OM is a matching with maximum weight. This implies that OM is also a cardinality matching. Hence, based on Definition 2, we have $|\mathrm{MM}| \geq |\mathrm{OM}|$.

Theorem 2:

Let $\mathcal{R}_1$ be the set of video clips retained by MM and $\mathcal{R}_2$ be the set of video clips retained by OM. Then

$$\mathcal{R}_2 \subseteq \mathcal{R}_1. \tag{11}$$

Proof: Denote p as the number of shots in a query clip, and $1 \leq \lambda \leq p$ is a parameter such that (p/λ) decides if a clip in the database should be filtered. If $|\mathrm{OM}| \geq (p/\lambda)$, then $|\mathrm{MM}| \geq (p/\lambda)$ since $|\mathrm{MM}| \geq |\mathrm{OM}|$ by Theorem 1. Since the clips whose matching shots are not less than (p/λ) are retained by MM and OM, $\mathcal{R}_2 \subseteq \mathcal{R}_1$.

In setting the hierarchical framework, λ is a parameter that controls the number of clips to be retained for OM. If the value of λ is large, the response time of a query will be slow. In our implementation, the λ is set to two, as mentioned in Section III-C.

B. Query-Dependent Video Clip Segmentation

In a video database, clips are not always available for retrieval. While shot boundaries can be readily located and indexed, clips boundaries are relatively harder to obtain since the detection of boundaries usually involves a certain degree of semantic understanding. The decomposition of videos into semantic clips is, in general, a hard problem. In this paper, instead of *explicitly* locating the boundaries of clips prior to video retrieval, we propose an *implicit* approach that exploits the inherent matching relationship between a given query and videos for online clip segmentation. Note that we do not need to "fix" the clip boundaries in videos for retrieval. The clips are indeed adaptively segmented during retrieval to meet the requirement of different queries. In other words, the results of clip segmentation can vary depending on the content of a query. Our approach is flexible in the sense that clip segmentation is query dependent, and, furthermore, no knowledge of clip boundaries is assumed before retrieval. The idea is similar to [5], where a sliding window (the window width depends on query length) is used for online clip identification. In our case, bipartite graph matching is exploited and 1-D clustering is used to extract potential clips with lengths that are independent of query length.

Given a query clip X and a video $\mathbf{Y}$ (usually $|\mathbf{Y}| \gg |X|$), a bipartite graph is constructed by matching the shots in X to the shots in $\mathbf{Y}$ by (4). The mapping is a many-to-many relationship, i.e., a shot can map to multiple shots in $\mathbf{Y}$ as long as they are considered similar based on the definition in (4). Denote $\zeta_j = \{0, 1\}$ to indicate whether a shot j in $\mathbf{Y}$ is matched by a shot in X. The mapping usually forms a number of dense and sparse clusters (with $\zeta_j = 1$ representing a match) along the 1-D space of ζ. The dense clusters indicate the presence of potentially similar video clips in $\mathbf{Y}$ with the query clip, while the sparse clusters can probably mean noisy matching.

One straightforward way of implicit clip segmentation is to extract the dense clusters directly from the 1-D ζ space. The procedure can be described as a 1-D connected-component grouping of the matched shot in $\mathbf{Y}$. To do this, we need two parameters (ρ, ϑ), where ρ specifies how to extract a cluster (or connected components) while ϑ specifies how to filter sparse clusters. The algorithm is formulated as follows. We check the distance d between all adjacent shots with $\zeta_j = 1$. Basically, d indicates the number of 0s between two neighboring 1s (or matched shots). All of the adjacent shots with $d \leq \rho$ are grouped into one cluster. In other words, the shot at the boundary of a cluster has at least $\rho + 1$ consecutive unmatched shots with other clusters. Once the clusters $Y_{k=\{0,1,\ldots\}}$ are extracted, we filter those clusters whose $|Y_k| < \vartheta$.

In the experiments, we set $\rho = 2$ and $\vartheta = (|X|)/(2)$. A large value of ρ can cause undersegmentation, while a small value of ρ can cause oversegmentation of video clips. The value of ρ is not easy to set, however, when $\rho = \{2, 3, 4, 5\}$, the setting mostly yields satisfactory results for our database of approximately 21 h of videos and 20 000 shots. The value of ϑ is set based on λ described in Theorem 2. Since $\lambda = 2$, any clip with $|Y_k| < (|X|)/(2)$ can never satisfy $|\mathrm{MM}| \geq (p/\lambda)$ and thus should not be considered.

A major advantage of our approach is that the segmentation is always tailored to the content of a query clip. Only those clips related to the query are segmented for retrieval. However, an implicitly segmented video clip may not be a precise scene or story since its boundary may contain shots from other clips and, furthermore, the clip itself could probably be composed of more than one clip due to undersegmentation. Some of these deficiencies, auspiciously, can be discarded during the similarity ranking of optimal matching. OM can be utilized not only to match similar shots, but also to split a clip and refine its boundary. Given a video clip $Y_k = \{y_1, y_2, \ldots, y_q\}$ and a query clip X, suppose only shots $\{y_\alpha, \ldots, y_\beta\}$ are matched with X, and $1 < \alpha < \beta < q$. The unmatched shots $Y_{k'} = \{y_1, \ldots, y_{\alpha-1}\}$ and $Y_{k'} = \{y_{\beta+1}, \ldots, y_q\}$ can be pruned if $|\alpha - 1| < (|X|)/(\lambda)$ and $|q - \beta| < (|X|)/(\lambda)$, respectively. Otherwise, $Y_{k'}$ and $Y_{k'}$ are split from Y_k as the new clips for similarity ranking by OM.

V. Video Summarization

One application of the clip-based similarity measure is event detection for video summarization. Here, we apply the proposed similarity measure for the clustering of news clips collected from news programs across different TV channels. The aim is to group clips which are under the same event but reported by various sources at different periods of time. Based on the outcome of clustering, video summaries which can include the highlighted events reported by different newscasts are automatically generated.

A. Graph-Based Clustering

Given a set of video clips, we model the similarity among clips as a weighted undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, where $\mathbf{V}$ is a set of video clips, and $\mathbf{E}$ is a set of edges that describes the proximity of clips. Our aim is to decompose $\mathbf{G}$ into subgraphs (or clusters) so as to minimize the intracluster distance while maximizing the intercluster distance. We adopt the normalized cut algorithm [22] for the recursive bipartitioning of $\mathbf{G}$ into the clusters of clips. Normalized cut aims to globally and optimally partition a graph $\mathbf{G}$ into two disjoint sets A and B ($A \cup B = \mathbf{V}$) by minimizing

$$N\mathrm{cut}(A,B) = \frac{\mathrm{cut}(A,B)}{\mathrm{volume}(A)} + \frac{\mathrm{cut}(A,B)}{\mathrm{volume}(B)} \tag{12}$$

where

$$\mathrm{cut}(A,B) = \sum_{i \in A, j \in B} \mathrm{Sim}_{\mathrm{OM}}(i,j) \tag{13}$$

$$\mathrm{volume}(A) = \sum_{i \in A, j \in \mathbf{V}} \mathrm{Sim}_{\mathrm{OM}}(i,j). \tag{14}$$

$\mathrm{cut}(A,B)$ is the sum of interclip similarity between A and B, $\mathrm{volume}(A)$ is the total similarity for all pairs of clips that connect A and $\mathbf{V}$, and $\mathrm{Sim}_{\mathrm{OM}}(i,j) = (\sum \omega_{ij})/(\min(p,q))$ is the similarity between clips i and j modified based on (5). The modification is to make $\mathrm{Sim}_{\mathrm{OM}}(i,j) = \mathrm{Sim}_{\mathrm{OM}}(j,i)$. $\mathrm{DP} + \mathrm{IF}$ is not used because it only affects the ranking but not the clustering of video clips. Equation (12) can be transformed to a standard eigen system

$$\mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}} z = \hat{\lambda} z \tag{15}$$

where $\mathbf{D}$ and $\mathbf{W}$ are $|\mathbf{V}| \times |\mathbf{V}|$ matrices. $\mathbf{D}$ is a diagonal matrix with $\mathbf{D}(i,i) = \sum_{j \in \mathbf{V}} \mathrm{Sim}_{\mathrm{OM}}(i,j)$ and $\mathbf{W}$ is a symmetrical matrix with $\mathbf{W}(i,j) = \mathrm{Sim}_{\mathrm{OM}}(i,j)$.

In (15), the eigen vector that corresponds to the second smallest eigen value is used to find the sets A and B. The value 0 is selected as the splitting point to divide the eigen vector into two parts that correspond to A and B, respectively. The algorithm runs recursively to further bipartitioning the resulting sets (or clusters). The procedure terminates when the average similarity for all pairs of video clips in a cluster is below $\mu + \alpha\sigma$, where μ and σ are the mean and standard deviation of all clip similarity in $\mathbf{G}$, respectively, and α is an empirical parameter.

B. Highlight Detection

Based on the event clusters obtained in Section V-A, highlight can be readily detected by selecting the representative clips from the clusters with large size. Assuming that the skimming time $\mathcal{S}$ of a summary is given, we use two heuristic criterion to select the highlight from clusters.

- *Cluster size*. Highlight events are usually repeatedly broadcast by different TV channels at different periods of time. Therefore, the number of times an event is broadcast is a vivid hint to deciding the highlight. Based on the skimming time constraint $\mathcal{S}$, we select the clusters for highlight summarization in the descending order of their cluster sizes.
- *Globality* of an event. An event that is broadcast by different TV channels is intuitively more important than an event that is broadcast by one channel only. Similarly, an event that is broadcast at different periods of time (e.g., morning, afternoon, and night) is more important than an event reported only at a particular time of day. Hence, we use these two hints (the number of channels and the number of periods) that an event is broadcast to decide the highlight, when the cluster sizes of two events are the same.

For each selected cluster $\mathcal{C}$, one representative clip is chosen for highlight summary. We select the clip (medoid) that is most centrally located in a cluster as representative. The medoid clip $\mathcal{M}_c$ is the clip whose sum of similarity with all other clips in its cluster is maximum, i.e.,

$$\mathcal{M}_c = \max_{i \in \mathcal{C}} \{\sum_{j \in \mathcal{C}} \mathrm{Sim}_{\mathrm{OM}}(i,j)\}. \tag{16}$$

VI. Experiments

A. Experiment I: Retrieval

To evaluate the performance of the proposed hierarchical framework for clip-based retrieval, we set up a database that consists of approximately 1272 min (more than 21 h) of videos. The genres of videos include news, sports, commercials, movies, and documentaries collected from different TV stations. In total, there are 19 929 shots. We conduct two

TABLE II
STATISTICS OF TESTING QUERIES

Query Type	# of queries	Average # of shot	Average # of relevant clip	Average difference of shot #	Average difference of frame #
Commercial	20	12.9	4.0	1.0	83
News	20	18.5	4.2	8.9	960
Sport	10	15.0	5.1	24.7	3010
Movie	5	16.0	5.2	23.9	1013.9
Documentary	5	11.4	3.0	6.7	775.4
Average	-	14.8	4.3	13.0	1168.5

TABLE III
COMPARISON BETWEEN LIU'S APPROACH AND OURS

	Liu's Approach	*Ours*
Features	color histogram, Tamura texture	color histogram
Video clips	manually segmented	online automatically segmented based on the content of query
Similarity factors	cluster-based matching, temporal order, speed, disturbance, congregate	optimal matching, temporal order, interference factor
Video clip filtering	linear combination, a manually optimized threshold is set to filter irrelevant clips	based on MM, $\lambda = 2$ as in Theorem 2
Video clip ranking	five weighting factors are manually optimized in the database	linear combination, three weights are set as in Eqn(9)

TABLE IV
EXPERIMENTAL RESULTS FOR VIDEO CLIP FILTERING AND RETRIEVAL

	Our Approach		Liu's Approach	
Query type	Precision	Recall	Precision	Recall
Commercial	0.935	1.000	0.628	0.990
News	0.794	0.735	0.649	0.622
Sport	0.765	0.684	0.601	0.509
Movie	0.617	0.626	0.364	0.405
Documentary	0.580	0.830	0.397	0.767
Average	0.738	0.775	0.528	0.659

experiments. The first one (cf. Section VI-A1) evaluates the filtering and ranking capability of our approach. We compare our approach with [16], which also considers different factors for clip retrieval. The second experiment (cf. Section VI-A2) assesses the performance of query-dependent segmentation and retrieval, and we compare our approach with [5], which also performs online clip segmentation and retrieval.

Various types of query are experimented for performance evaluation. These queries include clips from commercial, news, sport, movie and documentary videos. The relevancy of the commercial clips is based on the same products of the same companies. For news, the relevancy is based on the event. For sports, the relevancy is determined by the types of games being queried.[3] For movie and documentary clips, the relevancy is based on scenes with similar context. For each query, two assessors are involved in relevancy judgment. One assessor manually browses through the 21 h of videos and collects the relevant clips. The second assessor further verifies and confirms the relevancy of collected clips. In the experiments, since clips are segmented online, the results of segmentation will affect retrieval accuracy. In assessing the relevancy, a retrieved clip is judged to be relevant only if there is an overlap of at least 80% of shots with a relevant clip in the ground truth.

In total, 60 queries including clips from commercials, news, sports, movies, and documentaries are used for testing. Table II shows the statistics of the testing queries. The last two columns of Table II show the average difference of a query and a relevant clip in terms of the number of shots and frames. Compared with the average shot length of a query, there are 7.75%, 48.11%, 164.67%, 149.38%, and 58.77% differences for commercial, news, sports, movie, and documentary clips, respectively. This somewhat indicates the difficulty of retrieving clips where the shot lengths of certain video genres can be quite different.

1) Clip-Based Retrieval: We compare our approach with Liu's approach in [16]. As indicated in Table I, Liu's approach is similar to ours since various factors are taken into account for similarity measures of clips, although the underlying matching methodology and algorithms vary a lot. The major difference between these two approaches are summarized in Table III. In [16], a clustering-based algorithm is used to decide the matching of shots in two clips. The aim of the algorithm is to cluster the pairwise similarities of shots into the two groups which corre-

[3] The sport clips in our database are from news videos and weekly or daily reviews of sports games. Therefore, it is reasonable to judge the relevancy of sport clips by the type of game.

TABLE V
EXPERIMENTAL RESULTS FOR THE FILTERING AND RETRIEVAL OF NEWS CLIPS (MM + OM + DP + IF)

			Relevant	Our Approach		Liu's Approach	
	Query clip	Shot #	Clip #	Precision	Recall	Precision	Recall
1	Power cut accident in London	18	7	1.000	0.429	1.000	0.286
2	Bus bomb event in Israel	12	6	1.000	0.667	0.429	0.500
3	Six-way talk about North Korea	45	6	0.833	0.833	1.000	0.167
4	The death of an Iraq aga in bomb	29	6	0.714	0.833	1.000	0.667
5	New finance policy	33	6	0.500	0.833	0.800	0.667
6	UK premier follows investigation	15	5	0.200	0.600	0.167	0.400
7	Taiwan politic issue	22	4	1.000	0.750	0.600	0.750
8	National singing competition	22	4	1.000	0.750	1.000	0.750
9	Iraq Policy	10	4	1.000	0.500	0.235	1.000
10	Resignation of a UK official	11	4	0.500	0.500	0.250	0.500
11	CCTV program promotion	11	4	1.000	1.000	0.667	1.000
12	Chinese vice president meets foreigners	19	4	0.333	1.000	0.750	0.750
13	Iraq war	9	3	1.000	0.667	1.000	0.333
14	A UN official died in Iraq	17	3	1.000	0.667	0.222	0.667
15	New policies in the ministry of police	21	3	1.000	1.000	0.667	0.667
16	Soccer association election	16	3	1.000	0.667	0.400	0.667
17	Match for solar energy bus	23	3	1.000	0.667	0.500	0.333
18	Report about blaster virus	8	3	0.400	0.667	1.000	0.667
19	Conflict between Israel and Palestine	17	3	1.000	0.667	1.000	0.667
20	Intel CEO visits China	11	2	0.400	1.000	0.286	1.000
	Average	18.5	4.2	0.794	0.735	0.649	0.622

spond to the matched and unmatched shots. This is achieved by maximizing the centroid distance between two groups. Based on the matched shots, the temporal order, speed (duration difference), disturbance (number of unmatched shots), and congregation (number of one-to-one mapping) are computed for similarity measure. In our approach, the matching of shots and the degree of congregation are measured directly by OM. Dynamic programming is employed to measure the temporal order of two sequences. In [16], this value is measured by calculating the percentage of matching shots that are in reverse order. Our interference factor is the same as disturbance, and we do not use speed, since duration is not a critical factor in reflecting similarity, particularly when the unmatched shots are available. Notice that both approaches use different shot representation and similarity measurement. Our shot representation based on [20] is implemented to operate in a compressed domain and is not suitable for the extraction of Tamura features used in [16]. Basically, Liu's approach functions in uncompressed domain, and the shot similarity measurement is based on the weighted combination of color and Tamura features.

Liu's approach [16] assumes that video clips are presegmented and always available for retrieval. As a result, we manually segment the 21 h of videos into clips, and, in total, there are 1288 segmented video clips in our database. In the experiment, while the results of [16] are based on the retrieval of manually segmented video clips, our approach adopts the online automatic segmentation described in Section IV-B for retrieval.

TABLE VI
EXPERIMENTAL RESULTS FOR THE FILTERING AND RETRIEVAL OF SPORT CLIPS (MM + OM + DP + IF)

			Relevant	Our Approach		Liu's Approach	
	Query clip	Shot #	Clip #	Precision	Recall	Precision	Recall
1	Running	14	8	0.571	0.500	0.375	0.375
2	Swimming	8	7	0.600	0.857	0.429	0.429
3	Tennis	7	6	1.000	0.500	1.000	0.500
4	Gym	10	6	0.546	1.000	0.250	0.667
5	Judo	24	6	0.600	0.500	1.000	0.167
6	Boating	16	5	0.333	0.400	1.000	0.200
7	Diving	10	4	1.000	0.750	1.000	0.750
8	Basketball	22	3	1.000	1.000	0.500	1.000
9	Volleyball	19	3	1.000	0.667	0.333	0.667
10	Weight lifting	20	3	1.000	0.667	0.125	0.333
	Average	15	5.1	0.765	0.684	0.601	0.509

a) Video clip filtering: We use precision and recall to measure the performance. The recall and precision are defined as follows:

$$\text{Precision} = \frac{\text{Number of relevant clips being retained}}{\text{Number of clips being retained}}$$

$$\text{Recall} = \frac{\text{Number of relevant clips being retained}}{\text{Number of relevant clips}}.$$

In [16], no mechanism is proposed for the filtering of irrelevant clips. During the implementation, we set an optimized

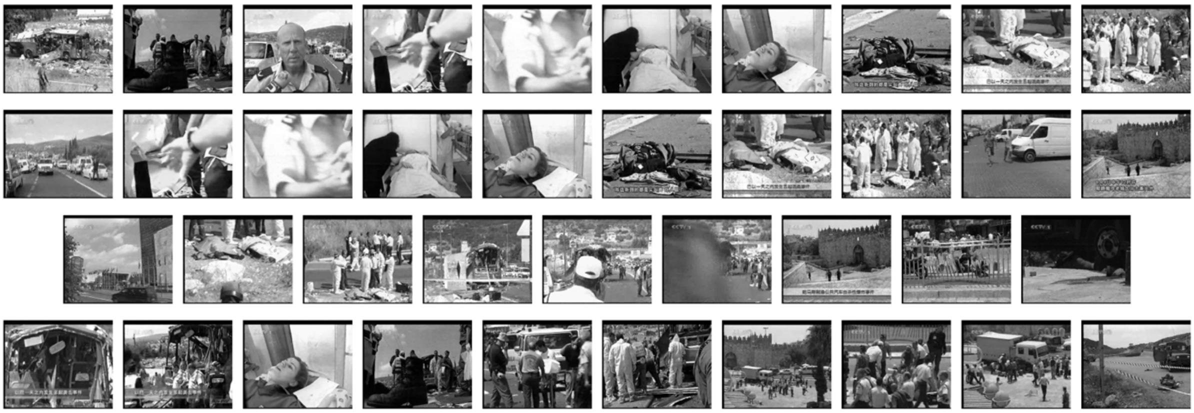


Fig. 2. Retrieval results of news query #2. Query clip is listed in first row. The correct matches are shown one row after another according to the ranked order.

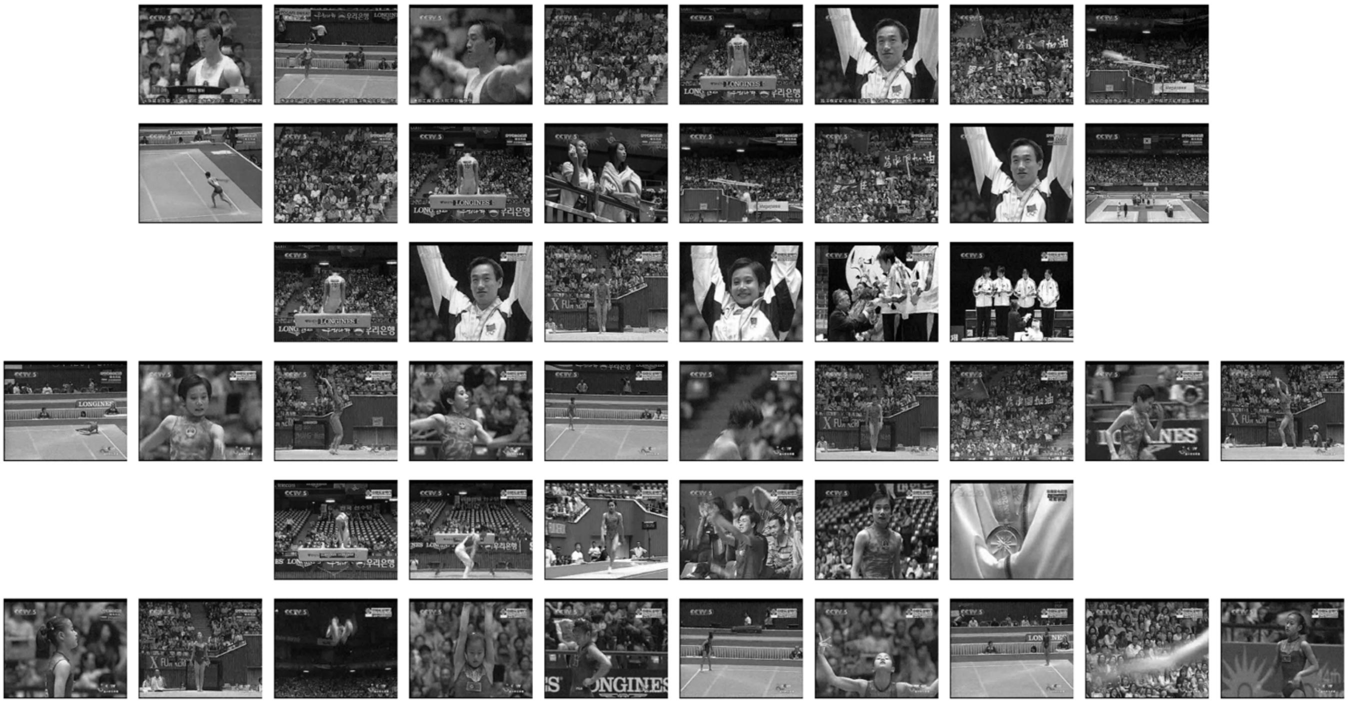


Fig. 3. Retrieval results of sports query #4. The query clip is listed in the first row. The correct matches are shown one row after another according to the ranked order.

threshold for this purpose. We systematically try different threshold values and select the one which gives the best overall recall and precision in our database as the threshold.

Table IV shows the experimental results of both approaches. The commercial retrieval is relatively easy since the visual content of relevant commercial clips is usually similar and the major differences are in temporal order and duration due to different shot composition. Both approaches achieve high recall, but our approach achieves better precision. Compared with commercial clips, the effective retrieval of news and sports video clips is harder because different newscasts tend to report the same event with different camera shootings and editions. In addition, more shots are generally included for a clip reported in a news program of longer duration. The details of news and sports queries are listed in Tables V and VI. Experimental results indicate that our proposed approach is, overall, superior to Liu's approach in terms of recall and precision, particularly in the retrieval of news and sports video clips collected from different TV channels. For movie and documentary queries, our approach shows constantly better

TABLE VII
EXPERIMENTAL RESULTS FOR VIDEO CLIP RANKING

Query type	Our Approach		Liu's Approach	
	AR	ANMRR	AR	ANMRR
Commercial	1.000	0.000	0.990	0.009
News	0.809	0.200	0.711	0.277
Sport	0.783	0.230	0.666	0.371
Movie	0.664	0.403	0.419	0.575
Documentary	0.833	0.160	0.767	0.214
Average	0.818	0.199	0.711	0.289

TABLE VIII
EXPERIMENTAL RESULTS FOR THE RANKING OF NEWS CLIPS (OM + DP + IF)

Query ID	Shot #	Relevant Clip #	Our Approach		Liu's Approach	
			AR	ANMRR	AR	ANMRR
1	18	7	0.714	0.231	0.571	0.528
2	12	6	1.000	0.136	0.500	0.457
3	45	6	1.000	0.000	0.667	0.284
4	29	6	0.667	0.284	0.833	0.161
5	33	6	0.667	0.321	0.833	0.198
6	15	5	0.800	0.243	0.400	0.557
7	22	4	0.750	0.224	0.750	0.224
8	22	4	1.000	0.052	0.750	0.224
9	10	4	0.750	0.259	1.000	0.000
10	11	4	0.500	0.466	0.500	0.466
11	11	4	1.000	0.000	1.000	0.000
12	19	4	1.000	0.000	0.750	0.224
13	9	3	0.667	0.364	0.667	0.303
14	17	3	0.667	0.303	0.667	0.303
15	21	3	1.000	0.000	0.667	0.303
16	16	3	1.000	0.000	0.667	0.303
17	23	3	0.667	0.303	0.667	0.394
18	8	3	0.667	0.303	0.667	0.303
19	17	3	0.667	0.303	0.667	0.303
20	11	2	1.000	0.200	1.000	0.000
Ave.	18.5	4.2	0.809	0.200	0.711	0.277

TABLE IX
EXPERIMENTAL RESULTS FOR THE RANKING OF SPORTS CLIPS (OM + DP + IF)

Query clip		Shot #	Relevant Clip #	Our Approach		Liu's Approach	
				AR	ANMRR	AR	ANMRR
1	Running	14	8	0.625	0.300	0.625	0.530
2	Swimming	8	7	0.857	0.165	0.714	0.341
3	Tennis	7	6	0.833	0.136	0.833	0.198
4	Gym	10	6	1.000	0.124	0.667	0.284
5	Judo	24	6	0.833	0.161	0.333	0.679
6	Boating	16	5	0.600	0.586	0.400	0.571
7	Diving	10	4	0.750	0.224	0.750	0.224
8	Basketball	22	3	0.667	0.303	1.000	0.061
9	Volleyball	19	3	1.000	0.000	0.667	0.303
10	Weight lifting	20	3	0.667	0.303	0.667	0.515
Average		15	5.1	0.783	0.230	0.666	0.371

performance. By manually investigating the retrieval results, we find that the advantage of our approach is mainly due to: 1) effectiveness of online clip segmentation in removing the sparse clusters of clips from graph matching and 2) capability of MM in filtering large amount of irrelevant clips.

Figs. 2 and 3 show the retrieval results of news query #2 and sports query #4, respectively (due to the limitation of space, we do not show all of the shots). Compared with commercial clips, the effective retrieval of news and sport clips is difficult, since the same event is usually reported in different profiles, editions, and camera angles, as shown in Figs. 2 and 3. Despite the difficulties, the proposed retrieval framework is still able to filter, match, and then rank the relevant clips with reasonably good accuracy.

b) Video clip ranking: In this experiment, our aim is to compare the ranking capability of both approaches. Specifically, the rank of each relevant clip is taken into account for performance evaluation, regardless of whether a clip is being filtered. Thus, MM is excluded from testing, otherwise relevant clips being filtered cannot be assessed. We use average recall (AR) and average normalized modified retrieval rank (ANMRR) [28] for performance evaluation. The values of AR and ANMRR range from $[0, 1]$. A *high* value of AR denotes the superior ability in retrieving relevant clips, while a *low* value of ANMRR indicates the high retrieval rate with relevant clips ranked at the top [28].

Table VII summarizes the experimental results while Tables VIII and IX show the details of news and sports retrieval. For the retrieval of commercial clips, both approaches attain almost perfect AR and ANMRR. This implies that all relevant clips are retrieved and ranked at the top. For the retrieval of clips from other video genres, our approach is constantly better than Liu's approach. By tracing the details of experimental results, we found that the cluster-based and temporal order algorithms used in Liu's approach cannot always give satisfactory results. In contrast, the proposed clip-based similarity can rank at least half of the relevant clips at the top-k ranked list (k depends on the number of relevant clips [28]).

The results in Table VII are based on the setting of $\alpha_{\mathrm{OM}} = 0.4$ (visual and granularity), $\alpha_{\mathrm{DP}} = 0.3$ (temporal order), and $\alpha_{\mathrm{IF}} = 0.3$ (interference). To investigate the effects of various factors, we conduct experiments to study the sensitivity of different settings for news, sports, movie, and documentary retrieval.[4] Table X shows the empirical results of varying the factor α_{OM} from 1.0 to 0.0. Basically, the performance is good when $\alpha_{\mathrm{OM}} > \alpha_{\mathrm{DP}} = \alpha_{\mathrm{IF}}$, and otherwise degrades. For the extreme case when $\alpha_{\mathrm{OM}} = 1.0$, the performance is indeed not as good as when $\alpha_{\mathrm{OM}} > \alpha_{\mathrm{DP}} = \alpha_{\mathrm{IF}} > 0$. When both $\alpha_{\mathrm{DP}} = \alpha_{\mathrm{IF}} = 0.3$,

[4] The effect of varying factors to commercial retrieval is insignificant, and thus the result is not shown.

TABLE X
SENSITIVITY ANALYSIS

Query	(0.4, 0.3, 0.3)		(0.8, 0.1, 0.1)		(1.0, 0.0, 0.0)		(0.2, 0.4, 0.4)		(0.0, 0.5, 0.5)	
type	AR	ANMRR	AR	ANMRR	AR	ANMRR	AR	ANMRR	AR	ANMRR
News	0.809	0.200	0.780	0.203	0.780	0.212	0.755	0.248	0.745	0.267
Sport	0.783	0.230	0.713	0.287	0.713	0.312	0.687	0.319	0.622	0.378
Movie	0.664	0.403	0.636	0.444	0.614	0.462	0.536	0.511	0.486	0.569
Documentary	0.833	0.160	0.767	0.220	0.767	0.220	0.667	0.314	0.667	0.344

The numbers in the bracket indicate the values of α_{OM}, α_{DP} and α_{IF} respectively.

TABLE XI
RESULT OF CLIP SEGMENTATION

Query	Our Approach			Chen's Approach [5]		
type	Average CP	Average FP	Average FN	Average CP	Average FP	Average FN
Commercial	11.850	0.325	0.150	10.500	2.432	4.250
News	20.213	1.788	1.939	16.375	5.473	6.283
Sport	22.691	1.310	1.357	17.250	3.230	5.348
Movie	33.937	3.188	10.188	26.731	8.500	14.235
Documentary	10.500	0.167	0.250	9.500	2.142	2.550
Average	19.838	1.356	2.777	16.071	4.355	6.533

CP: correct positive; FP: false positive; FN: false negative

the best performance is achieved. Overall, by attempting different settings, the results (except documentary queries) are still better than Liu's approach, except when $\alpha_{\text{OM}} = 0$.

Currently, on a Pentium-M 1.5-GHz machine with 512-M memory, the average retrieval time for a query by using OM + DP + IF is approximately 1.639 s. If MM + OM + DP + IF is used, the average retrieval time is 0.971 s. Although the MM and OM are not linear time algorithms, they are still very efficient even in a large database, since the online segmentation (linear time algorithm) has removed large portions of video segments from consideration before MM and OM matching.

3) Query-Dependent Clip Segmentation and Retrieval: Even though the retrieved clips by our approach are segmented online, the boundaries of most clips are precisely located, particularly when OM is used to refine the boundaries of the retrieved clips. To verify the performance, we count the number of falsely included (false positive) and missed (false negative) shots over all of the retrieved and relevant clips of 60 queries. In addition, we compare our approach with Chen and Chua's approach in [5], which also performs query-dependent clip segmentation. In [5], a sliding window with length (in terms of shot number) set to be 10% larger than the query is moved along the time axis to locate the potential candidates. Table XI summarizes the performance of segmentation in terms of recall and precision. For our approach, only a few oversegmentation and undersegmentation of clips happen. On average, each retrieved clip has at most two falsely included and three missed shots. Overall, our approach achieves precision of 95% and recall of 92%, compared with Chen's approach with 80% of precision and 73% recall. Our approach outperforms Chen's method in all video genres we have tested.

To contrast both approaches in term of retrieval effectiveness, we conduct another experiment using the same set of 60 queries. Since the aim of this experiment is to assess the performance of retrieval with different segmentation schemes, we implement [5] with the same key-frame representation and shot similarity measurement as ours. Table XII shows the experimental results. Overall, our approach constantly show better performance than [5]. Chen's approach shows competitive results for matching identical or almost identical clips (e.g., commercials), but not for clips that have undergone different capturing and editing effects.

B. Experiment II: Summarization

We use 10 h of videos for testing. These videos are recorded continuously for four days from seven different TV channels. There are a total of 40 different news programs with durations ranging from 5 to 30 min. As observed from these videos, the same events are repeatedly broadcast in different editions and profiles by different stations. Even the same event reported at the same channel can be shown differently at different times of reporting.

We manually segment the videos into clips. In total, there are 439 news clips. The number of events that are reported more than once are summarized in Table XIII. In total, there are 115 clips involved in reporting 41 events. Our aim is to group news clips that describe the same event under a cluster and then select the clusters as well as the representatives of clusters for summarization.

1) Clustering: We employ F-measure [23] to evaluate the performance of video clip clustering. F-measure evaluates the quality of clusters by comparing the detected and ground-truth

TABLE XII
PERFORMANCE COMPARISON WITH CHEN'S APPROACH

Query	Our Approach				Chen's Approach [5]			
type	Precison	Recall	AR	ANMRR	Precision	Recall	AR	ANMRR
Commercial	0.935	1.000	1.000	0.000	0.881	0.965	1.000	0.024
News	0.794	0.735	0.809	0.200	0.658	0.559	0.649	0.406
Sport	0.765	0.684	0.783	0.230	0.482	0.402	0.528	0.511
Movie	0.617	0.626	0.664	0.403	0.569	0.476	0.433	0.517
Documentary	0.580	0.830	0.833	0.160	0.453	0.767	0.767	0.270
Average	0.738	0.775	0.818	0.199	0.609	0.634	0.675	0.346

clusters. Letting $\mathcal{G}$ be the set of ground-truth clusters and $\mathcal{D}$ be the set of detected clusters, the F-measure $\mathbf{F}$ is given as

$$\mathbf{F} = \frac{1}{\mathcal{Z}} \sum_{\mathcal{C}_i \in \mathcal{G}} |\mathcal{C}_i| \max_{\mathcal{C}_j \in \mathcal{D}} \{\mathcal{F}(\mathcal{C}_i, \mathcal{C}_j)\} \tag{17}$$

$$\mathcal{F}(\mathcal{C}_i, \mathcal{C}_j) = \frac{2 \times \text{Recall}(\mathcal{C}_i, \mathcal{C}_j) \times \text{Prec}(\mathcal{C}_i, \mathcal{C}_j)}{\text{Recall}(\mathcal{C}_i, \mathcal{C}_j) + \text{Prec}(\mathcal{C}_i, \mathcal{C}_j)} \tag{18}$$

where

$$\text{Recall}(\mathcal{C}_i, \mathcal{C}_j) = \frac{|\mathcal{C}_i \cap \mathcal{C}_j|}{|\mathcal{C}_i|} \tag{19}$$

$$\text{Prec}(\mathcal{C}_i, \mathcal{C}_j) = \frac{|\mathcal{C}_i \cap \mathcal{C}_j|}{|\mathcal{C}_j|}. \tag{20}$$

The term $\mathcal{Z} = \sum_{\mathcal{C}_i \in \mathcal{G}} |\mathcal{C}_i|$ is a normalization constant. The value of $\mathbb{F}$ is in the range $[0, 1]$, and $\mathbf{F} = 1$ indicates perfect clustering. By the normalized cut algorithm and clip-based similarity, we detect 291 clusters in the 10 h of videos. The value of F-measure is $\mathbf{F} = 0.8225$, where $|\mathcal{G}| = 290$ and $|\mathcal{D}| = 291$. Table XIV shows the details of few clustering results. Some clusters such as events #1 and #3 are oversegmented into two clusters, respectively. Some false clips are included due to the similarity in background color, but no relevant clip is missed. Because we select the medoid of a cluster as representative, false clips are not selected in video summaries. Fig. 4 shows the clustering result of event #6 in Table XIV. Our approach successfully groups the three video clips in one cluster although they are from three different TV channels and appear differently.

2) News Summation: Given a skimming time, our approach selects clusters based on the cluster size and globality of events. The medoids of selected clusters are then included in the summary. The ground-truth summary is manually generated in the same way based on the ground-truth clusters. For instance, when the skimming time equals 10 min, the ground-truth summary includes all three events that are broadcast six times and the other three events that are reported four times (see Table XIII). Table XV shows the results of summarization. Experimental results indicate that our approach can include most of the expected events for summarization. Some events are repeated due to the oversegmentation of clusters.

C. Discussion

1) Retrieval by Compact Representation Beyond Shot Structure: Our clip similarity measurement is based on the matching of information at the shot level. One practical question is the effectiveness of retrieval compared with those that perform matching grounded on compact video representation that is beyond shot structure. Possible ways of compact representation include summarization of clip information and clustering of clips prior to retrieval. Both approaches, particularly the latter one, can speed up the matching time, at the expense of requiring the segmentation of clips in advance. The deployment of compact representation beyond shot relies on the argument of whether to perform clip segmentation before or during retrieval.

TABLE XIII
NUMBERS OF NEWS EVENTS THAT ARE BROADCAST MORE THAN ONE TIME

Broadcast #	Number of events
6	3
4	5
3	11
2	22

Segmentation of clips, in general, is task-, domain-, and even user-dependent. Clip segmentation has been studied by [2], [3], [10], [27] but with varying clip definitions. Although accurate segmentation results are shown, they usually depend on the types of videos being used. Indeed, it is not easy to find a concise clip definition to be shared by different video genres. For video retrieval across genres, our query-dependent segmentation strategy alleviates the need for segmentation in advance and delays the segmentation decision until the availability of query requirement. With our approach, speed may be a concern since shot-to-shot similarity comparison is required during retrieval. Nevertheless, owing to the fact that we use a compact representation for shot content, the comparison between shots is indeed fast. Although clustering of clips can provide better speed efficiency, it does have the disadvantage that any mistake made during segmentation may not easily be corrected during retrieval. Our proposed clip segmentation and filtering techniques are suitable for retrieval tasks where the clip cannot be readily defined before retrieval and when cluster-based clip indexing is absent.

2) Multiple Features: In videos, there are multiple features (e.g., color, motion, duration, average shot length, shot tempo, audio, and caption) that can be jointly utilized to improve the retrieval accuracy of our approach. In this paper, we consider

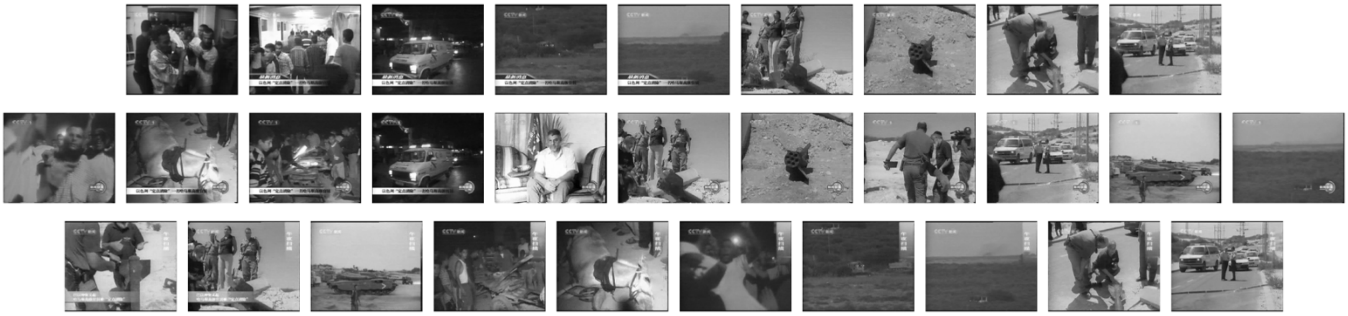


Fig. 4. Clustering results of event #6 in Table XIV. The three news clips are clustered correctly. The cluster medoid is listed in the second row.

TABLE XIV
CLUSTERING RESULTS OF SOME NEWS EVENTS

	News event	Number of clips in the event	Average number of shots	Final cluster(s)	Falsely included clips
1	Six-way talk about North Korea	6	55	2	2
2	New financial policy	6	22	1	2
3	The death of an Iraq aga in bomb	6	21	2	0
4	A conflict event in Iraq	4	15	1	2
5	Economic development of Beijing	4	8	1	1
6	Conflict between Israel and Palestine	3	11	1	0
7	Report about blaster virus	3	6	1	0

TABLE XV
RESULTS OF SUMMARIZATION FROM VIDEOS OF 10 H

Skimming time (Minute)	Number of Expected events (Ground-truth)	Number of clips included in summary	Detected events	Missed events	Repeated events
10	6	8	4	2	0
20	11	14	8	3	0
30	24	26	21	3	0
40	39	39	31	8	1
45	41	42	34	7	2

only color information, although other features can also be incorporated into the current framework. When multiple features are under sail, perhaps an associated problem is how to fuse or integrate multiple information for more effective retrieval. Since this paper focuses on the clip similarity measure, the issue of multifeature integration is not addressed. Nonetheless, to show the effectiveness of our framework with multiple features, we conduct a simple experiment to investigate retrieval performance when color and motion features are utilized. For each shot, in addition to color feature, an eight-bin motion directional histogram is generated by extracting motion vectors in the compressed domain. Histogram intersection is used as the distance measure for motion features. The color and motion cues are fused linearly, with color having a weight of 0.7 and motion with a weight of 0.3. Table XVI shows the experimental results. Overall, the ranking of news and sport clips improves, while there is no change for commercial clips, when both color and motion features are incorporated.

3) Matching Effectiveness: There might be multiple shots in clip Y that are similar to one shot in clip X. For this case, the use of IF under one-to-one matching may not be a fair strategy. To tackle this problem, one possible way is to allow many-to-many (M2M) matching. Nevertheless, the disadvantage of M2M is that false matches can easily be included during matching. M2M has the potential to become a good scheme for retrieval, but only if there exists a good mechanism for reducing false matches when granting one shot to match multiple shots. Under the constraint of one-to-one mapping, the role of interference is to downgrade similarity for clips with unmatched shots. Obviously, it is not fair to penalize the true but unmatched shots, but in practice there is no good solution to decide which shots are actually "true but unmatched." In our approach, even though IF is considered, it only affects similarity ranking but not overall recall when clip filtering is used.

On the other hand, we might have the case where two clips are similar but with only a few shots being matched, e.g., two

TABLE XVI
RETRIEVAL RESULTS FOR MULTIPLE FEATURES

Query type	# of queries	Color + Motion		Color	
		AR	ANMRR	AR	ANMRR
Commercial	20	1.000	0.000	1.000	0.000
News	20	0.847	0.151	0.809	0.200
Sport	10	0.842	0.173	0.783	0.230
Average	-	0.896	0.108	0.864	0.143

clips taken in the same place but in different parts of the location, with only a few shots sharing similar visual information. Owing to the lack of visual evidence, our clip filtering discards the clips from further consideration. Generally speaking, this is an extremely difficult case, considering the fact that most shot contents in two clips appear differently even when they are captured in the same physical location. One plausible way to solve this problem is to reconstruct the location information of a clip for matching, as attempted by Aner in [2] and [3]. Nevertheless, the reconstruction of location information is appropriate perhaps for certain kinds of dramas (for example, sitcoms used in [2] and [3]), where one scene usually takes place in one location. In general, location reconstruction is not trivial when considering videos across multiple genres.

VII. CONCLUSION

We have presented the proposed approach and algorithm for clip-based similarity measure. Two major applications: hierarchical video retrieval and video summarization, based on the similarity measure, have also been described and experimented. Encouraging results have been obtained through the performance evaluation in two databases with 21 and 10 h of videos, respectively. Experimental results suggest that the proposed MM is effective in filtering irrelevant clips, while OM is capable of effectively retrieving and clustering video clips of same event.

Although the current clip-based similarity measure considers only color features, other features such as motion, audio, and caption can also be incorporated in the existing framework to improve the retrieval accuracy. Currently, the implementation of MM and OM is based on the Kuhn and Kuhn–Munkres algorithms, which require $O(nm)$ and $O(n^4)$, respectively, where n and m are the number of shots and matching edges. Faster versions of MM and OM exist [21]; for instance, the computational complexity of MM can be as fast as $O(\sqrt{n}m)$ and OM can be as fast as $O(n(m + n\log n))$. In addition, approximate versions of MM and OM algorithms do exist [4], [21] that provide another glimpse of opportunity to compromise the tradeoff between speed and retrieval effectiveness.

REFERENCES

[1] D. A. Adjeroh, M. C. Lee, and I. King, "A distance measure for video sequences," *Comput. Vis. Image Understanding*, vol. 75, no. 1–2, pp. 25–45, Jul.–Aug. 1999.

[2] A. Aner and J. R. Kender, "Mosaic-based clustering of scene locations in videos," in *Proc. IEEE Workshop Content-Based Access of Image and Video Libraries*, Dec. 2001, pp. 111–117.

[3] A. Aner and J. R. Kender, "Video summaries through mosaic-based shot and scene clustering," in *Int. Conf. European Conf. on Computer Vision*, May 2002.

[4] D. P. Bertsekas, "Auction algorithms for network flow problems: A tutorial introduction," *Computat. Optimization Applicat.*, vol. 1, pp. 7–66, 1992.

[5] L. Chen and T. S. Chua, "A match and tiling approach to content-based video retrieval," in *Proc. Int. Conf. Multimedia and Expo*, 2001, pp. 417–420.

[6] S. C. 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.

[7] S. C. Cheung and A. Zakhor, "Fast similarity search and clustering of video sequences on the world-wide-web," *IEEE Trans. Multimedia*, vol. 7, no. 3, pp. 524–537, Jun. 2005.

[8] T. S. Chua, S. F. Chang, L. Chaison, and W. Hsu, "Story boundary detection in large broadcast news video archives—Techniques, experience and trends," in *Proc. ACM Multimedia Conf.*, 2004, pp. 656–659.

[9] N. Dimitrova and M. Abdel-Mottaled, "Content-based video retrieval by example video clip," in *SPIE Proc. Storage and Retrieval of Image and Video Database VI*, 1998, vol. 3022, pp. 184–196.

[10] A. Hanjalic, R. L. Lagendijk, and J. Biemond, "Automated high-level movie segmentation for advanced video retrieval systems," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 5, pp. 580–88, Jun. 1999.

[11] T. C. Hoad and J. Zobel, "Fast video matching with signature alignment," in *Proc. Int. Workshop Multimedia Inf. Retrieval*, 2003, pp. 262–268.

[12] A. K. Jain, A. Vailaya, and W. Xiong, "Query by video clip," *ACM Multimedia Syst. J.*, vol. 7, pp. 369–384, 1999.

[13] K. Kashino, T. Kurozumi, and H. Murase, "A quick search method for audio and video signals based on histogram pruning," *IEEE Trans. Multimedia*, vol. 5, no. 3, pp. 348–357, Sep. 2003.

[14] J. R. Kender and B. L. Yeo, "Video scene segmentation via continuous video coherence," in *Proc. Int. Conf. Comput. Vis. Pattern Recognit.*, 1998, pp. 367–373.

[15] R. Lienhart and W. Effelsberg, "VisualGREP: A systematic method to compare and retrieve video sequences," *Multimedia Tools Applic.*, vol. 10, no. 1, pp. 47–72, Jan. 2000.

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

[17] L. Lovasz and M. D. Plummer, *Matching Theory*. Amsterdam, The Netherlands: North Holland, 1986.

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

[19] C. W. Ngo, T. C. Pong, and R. T. Chin, "Video partitioning by temporal slices coherency," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 11, no. 8, pp. 941–953, Aug. 2001.

[20] C. W. Ngo, T. C. Pong, and H. J. Zhang, "Motion-based video representation for scene change detection," *Int. J. Comput. Vis.*, vol. 50, no. 2, pp. 127–143, 2002.

[21] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*. Berlin, Germany: Springer-Verlag, 2003, vol. A, pp. 267–290.

[22] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[23] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *Proc. KDD Workshop Text Mining*, 2000, pp. 1–20.

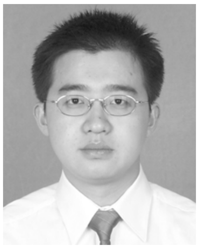
[24] Y. P. Tan, S. R. Kulkarni, and P. J. Ramadge, "A framework for measuring video similarity and its application to video query by example," in *Proc. Int. Conf. Image Process.*, 1999, pp. 106–110.

[25] Y. Wu, Y. Zhuang, and Y. Pan, "Content-based video similarity model," in *Proc. ACM Multimedia Conf.*, 2000, pp. 465–467.

[26] W. S. Xiao, *Graph Theory and Its Algorithms*. Beijing, China: Aviation Industry, 1993.

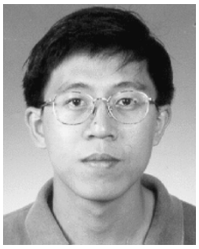
[27] M. M. Yeung and B. L. Yeo, "Video visualization for compact presentation and fast browsing of pictorial content," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 5, pp. 771–785, Oct. 1997.

[28] *Description of Core Experiments for MPEG-7 Color/Texture Descriptors*, ISO/MPEGJTC1/SC29/WG11 MPEG98/M2819, MPEG video group, Jul. 1999.



Yuxin Peng received the B.Eng. degree in computer science and technology from Fuzhou University, Fujian, China, in 1997, the M.Eng. degree in computer engineering from Guizhou University of Technology, Guizhou, China, in 2000, and the Ph.D. degree in computer science and technology from Peking University, Beijing, China, in 2003.

He joined the Institute of Computer Science and Technology, Peking University, as an Assistant Professor in 2003 and was promoted to Associate Professor in 2005. From 2003 to 2004, he was a Visiting Scholar with the Department of Computer Science, City University of Hong Kong. His current research interests include content-based video retrieval, image processing, and pattern recognition.



Chong-Wah Ngo (M'02) received the B.Sc. and M.Sc. degrees from Nanyang Technological University of Singapore in 1994 and 1996, respectively, both in computer engineering, and the Ph.D. degree in computer science from the Hong Kong University of Science and Technology, Kowloon, Hong Kong, in 2000.

Before joining the City University of Hong Kong as Assistant Professor with the Computer Science Department in 2002, he was a Postdoctoral Scholar with the Beckman Institute, University of Illinois, Urbana-Champaign. He was also a Visiting Researcher with Microsoft Research Asia in 2002. His research interests include video computing, multimedia information retrieval, data mining, and pattern recognition.