UQLIPS: A Real-time Near-duplicate Video Clip Detection System

Heng Tao Shen Xiaofang Zhou Zi Huang Jie Shao Xiangmin Zhou
School of Information Technology and Electrical Engineering
The University of Queensland
{shenht, zxf, huang, jshao, emily}@itee.uq.edu.au

ABSTRACT

Near-duplicate video clip (NDVC) detection is an important problem with a wide range of applications such as TV broadcast monitoring, video copyright enforcement, content-based video clustering and annotation, etc. For a large database with tens of thousands of video clips, each with thousands of frames, can NDVC search be performed in real-time? In addition to considering inter-frame similarity (i.e., spatial information), what is the impact of frame sequence similarity (i.e., temporal information) on search speed and accuracy? UQLIPS is a prototype system for online NDVC detection. The core of UQLIPS comprises two novel complementary schemes for detecting NDVCs. Bounded Coordinate System (BCS), a compact representation model ignoring temporal information, globally summarizes each video to a single vector which captures the dominating content and content changing trends of each clip. The other proposal, named FRAme Symbolization (FRAS), maps each clip to a sequence of symbols, and takes temporal order and sequence context information into consideration. Using a large collection of TV commercials, UQLIPS clearly demonstrates that it is feasible to perform real-time NDVC detection with high accuracy.

1. INTRODUCTION

An important research issue in multimedia databases is fast and robust content-based video retrieval (CBVR) in large video collections [3, 5, 10]. A special problem of CBVR is near-duplicate video clip (NDVC) detection, which searches for the near-duplicates of a query clip. Video clips are defined as *short clips in video format*. Unlike traditional long videos such as TV programs and full movies, video clips are mostly less than 10 minutes and overwhelmingly supplied by amateurs. The widespread popularity of video clips, with the aid of WWW, has evolved into *clip culture*. Extending the definition of near-duplicate images [6, 8], we define NDVCs as video clips that are similar or nearly duplicate of each other, but appear differently due to various changes introduced during capturing time (camera view point and setting, lighting condition, background, foreground, etc.), transformations (video format, frame rate, resize, shift, crop, gamma, contrast, brightness, saturation, blur, age, sharpen, etc.), and editing operations (frame insertion, deletion, swap and content modification).

NDVC detection has a wide range of applications such as TV broadcast monitoring, copyright enforcement, online video usage monitoring, video database purge, video clustering and annotation, cross-modal divergence detection, etc. Consider an application of NDVC detection in TV broadcast monitoring. When a company contracts TV stations for certain commercials, it often asks a market survey company to monitor whether its commercials are actually broadcasted as contracted. These market survey companies are often approached by other companies who are interested in understanding how their competitors conduct advertisements. While the same commercial is given to all TV stations, it can be broadcasted with some variations, such as TV station-specific parameters (e.g., frame rate, aspect ratio, gamma and resolution), TV reception and recording errors (on signal quality and color degradation), and inserts of different products or contact information (e.g., a supermarket wants to insert different products on sale in the same TV commercial template). Thus, the 'same' TV commercial broadcasted by different TV stations at different time are NDVCs. Also, the increasing generation and dissemination of video clips have created an urgent need for video search engines to facilitate finding and browsing relevant clips. According to a July 16, 2006 announcement by YouTube [1], a popular free video sharing web site that lets users upload, and view video clips, about '100 million clips are viewed daily on YouTube, with an additional 65,000 new video clips uploaded per 24 hours'. An important problem faced by video search engines now is how to perform fast video clip search for a new clip from their huge collections to avoid copyright violation and perform database purge.

Due to the high complexity of video features (e.g., a sequence of high-dimensional frame vectors), real-time NDVC detection from large video databases is very challenging. UQLIPS, a fast and robust NDVC detection system, is developed to demonstrate that NDVC detection can be performed, using the novel search methods we proposed, fast enough to support real-time search in a large video clip database. We also demonstrate the differences on query speed and accuracy for two different categories of search methods, one considers video temporal information and the other does not. The state-of-the-art methods, Edited Dis-

VLDB '07, September 23-28, 2007, Vienna, Austria.

tance on Real sequence (EDR) [4] and Video Triplet (ViTri) [7], are selected as the baseline for each category. Edit distance is extensively used to take temporal constraint and alignment into consideration for string and biological sequence matching. It can be used to measure the minimum number of atomic edit operations (insertions, deletions, and substitutions) needed to transform one video sequence to another [2]. In our early work [7], neglecting the temporal information of video, similar frames can be summarized into a single cluster, which is modelled by a tightly bounded hyper-sphere described by its position, radius and density (ViTri). Video similarity is then estimated by the volumes of intersection between hyper-spheres multiplying the minimal density. In this demonstration, we show two novel video representation models, BCS and FRAS for NDVC search. For each clip in video database, BCS statistically summarizes the intrinsic distribution of all frame points in feature vector space into a single vector. This compact representation reduces the size of video data dramatically and the complexity of its similarity measure is only linear in the dimensionality of feature space (independent of video length). For taking temporal information into consideration, we present another strategy, FRAS, which is based on frame symbolization. For each video clip, FRAS representation can capture not only its inter-frame similarity information but also sequence context information. FRAS employs effective methods to compensate the information loss caused by frame symbolization to ensure high accuracy in NDVC search. From a database of tens of thousands of TV commercials, with UQLIPS we demonstrate that BCS can achieve high quality retrieval with response time only in milliseconds, while FRAS achieves robust matching with response time typically in seconds. We show that ViTri suffers from poor accuracy due to its approximation nature, and EDR, while also accurate, is several times slower than FRAS.

2. THE SYSTEM

2.1 System Architecture

Figure 1 shows the system architecture of UQLIPS. We demonstrate the functionalities of UQLIPS with a large video collection consisting of TV commercials, which are captured from free-to-air TV broadcasting and pre-processed offline to create the database. Live digital TV broadcasting is continuously captured (from different TV stations at different frame rates and resolutions). The commercial breaks are identified automatically and stored as individual clips in the database. Through image feature extraction, each clip is represented by a sequence of high-dimensional frame feature vectors. Our system can support multiple kinds of features such as RGB and HSV color histograms. This process captures both spatial and temporal information inherent in video clips. The extracted visual content features are then compacted using four different schemes: two from the category that retains temporal information and two from the category that does not. The two methods selected for each category include one state-of-the-art method in the category (ViTri [7] and EDR [4]), and one new method developed by the authors (BCS and FRAS, to be described next).

UQLIPS performs online near-duplicate search. That is, for a user-submitted query clip, the system extracts the features from the clip using the same process used to generate the database, and then performs a k nearest neighbor similarity search. In UQLIPS, the compact video summarizations are all indexed to reduce the search space, using either the indexing methods introduced in the original papers (for EDR and ViTri), or the methods specifically designed for the two new methods. The indexing methods mainly differ from the transformation methods used, but are all based on B$^+$-tree index after transformation.

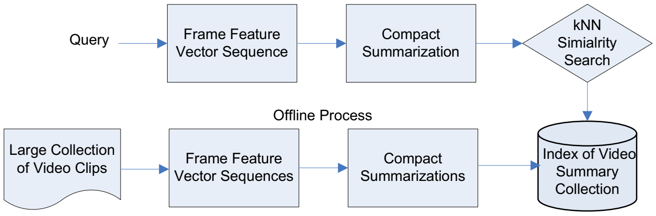

Query
Frame Feature Vector Sequence
Compact Summarization
kNN Similarity Search
Offline Process
Large Collection of Video Clips
Frame Feature Vector Sequences
Compact Summarizations
Index of Video Summary Collection

Figure 1: System Architecture of UQLIPS.

2.2 BCS

First, we introduce BCS, which is a novel single video representation model aiming at capturing the dominating content and content changing trends of video. Given the frame point distribution in high-dimensional feature vector space of each video clip, BCS represents it by a new coordinate system, where each of its coordinate axes is identified by the directions of great variances. These coordinate axes are all mutually orthogonal at the origin, which is the mean of all frame points of the clip. Here, we explicitly use a bounded scheme to identifies a line segment bounded by two furthermost projections to capture the range of data projections along each axis of new coordinate system. Therefore, these identified line segments indicate the ranges of frame feature vector distribution along certain orientations of a video clip.

Given a video clip $X = \{x_1, x_2, \ldots, x_n\}$, where x_i is a d-dimensional feature vector (normally $n \gg d$), its Bounded Coordinate System $BCS(X) = (O, \ddot{\Phi}_1, \ldots, \ddot{\Phi}_d)$ is determined by the mean for all x_i denoted as O and d orientations and ranges. Independent of the frame number n, BCS only records a origin and d identified line segments to represent a clip. A BCS actually consists of $(d + 1)$ d-dimensional points, and it is a global summarization that captures the dominating content and content changing trends of a video clip.

Given videos X and Y and their compact summarizations $BCS(X) = (O^X, \ddot{\Phi}_1^X, \ldots, \ddot{\Phi}_d^X)$ and $BCS(Y) = (O^Y, \ddot{\Phi}_1^Y, \ldots, \ddot{\Phi}_d^Y)$, where d is the numbers of identified line segments of new coordinate system (or space dimensionality), their video similarity is estimated by the similarity between their BCSs. In each BCS, origin measures the average position of all points, and identified line segments indicate the directions of large variances together with the dispersions of data projections. Two BCSs can be matched by performing translation, rotation and scaling operations. A *translation* allows one to move its origin to another position ($||O^X - O^Y||$). A *rotation* defines an angle which specifies the amount to rotate an axis to match its counterpart in another BCS. A *scaling* operation can stretch or shrink an axis to be of equal length to another. In vector space, the difference of two vectors is given by the length of their substraction, which nicely takes both rotation and scaling operations into consideration to match two corresponding line segments ($||\ddot{\Phi}_i^X - \ddot{\Phi}_i^Y||$). Compared with the quadratic time complexity of existing methods, the time complexity of BCS similarity measure is linear.

2.3 FRAS

BCS is a single video representation for each clip. However, this summarization neglects temporal order inherent in video sequences. FRAS, a symbolization based technique, preserves temporal order of video sequences for more actuate NDVC search [9].

FRAS first produces a *symbol dictionary* by performing hierarchical clustering (e.g., k-means) over the whole frame dataset. Each item in this dictionary is a small frame cluster whose radius is not greater than ϵ, which is a threshold for frame similarity. A cluster C is denoted as $< c, O, r, N >$, where c is the cluster id, O and r are the cluster center and radius, and N is the number of frames in the cluster.

Given a video clip $X = \{x_1, x_2, \ldots, x_n\}$, for each frame x_i, by looking up the symbol dictionary and checking the clusters containing it, the mapping from x_i to c_i can be done easily. If no cluster containing x_i is found in the dictionary, a special symbol '-' is used to represent x_i. In this way, the video sequence $X = \{x_1, x_2, \ldots, x_n\}$ is then transformed into a symbol sequence $S = \{c_1, c_2, \ldots, c_n\}$. Compared with the dimension-wise quantization based symbolization method [2], FRAS summarization is more compact, since it represents each frame by a cluster id instead of a symbol string. Transforming each video frame into a symbol inevitably incurs some information loss, since the neighboring clusters in high-dimensional space are usually overlapped with each other. The information loss can be reduced by representing each frame of query by multiple symbols that contain the frame. Also, we believe that this information loss can be further compensated by context information of symbol sequence matching.

Given two video symbol sequences S and $S^{'}$ of length m and n respectively, their similarity can be measured by Probability-based Edit Distance (PED), which extends string edit distance. Let S_{m-1} be the subsequence of the first $m-1$ elements of S, and $S[i]$ be the i^{th} element of S, $PED(S, S^{'})$ is defined as:

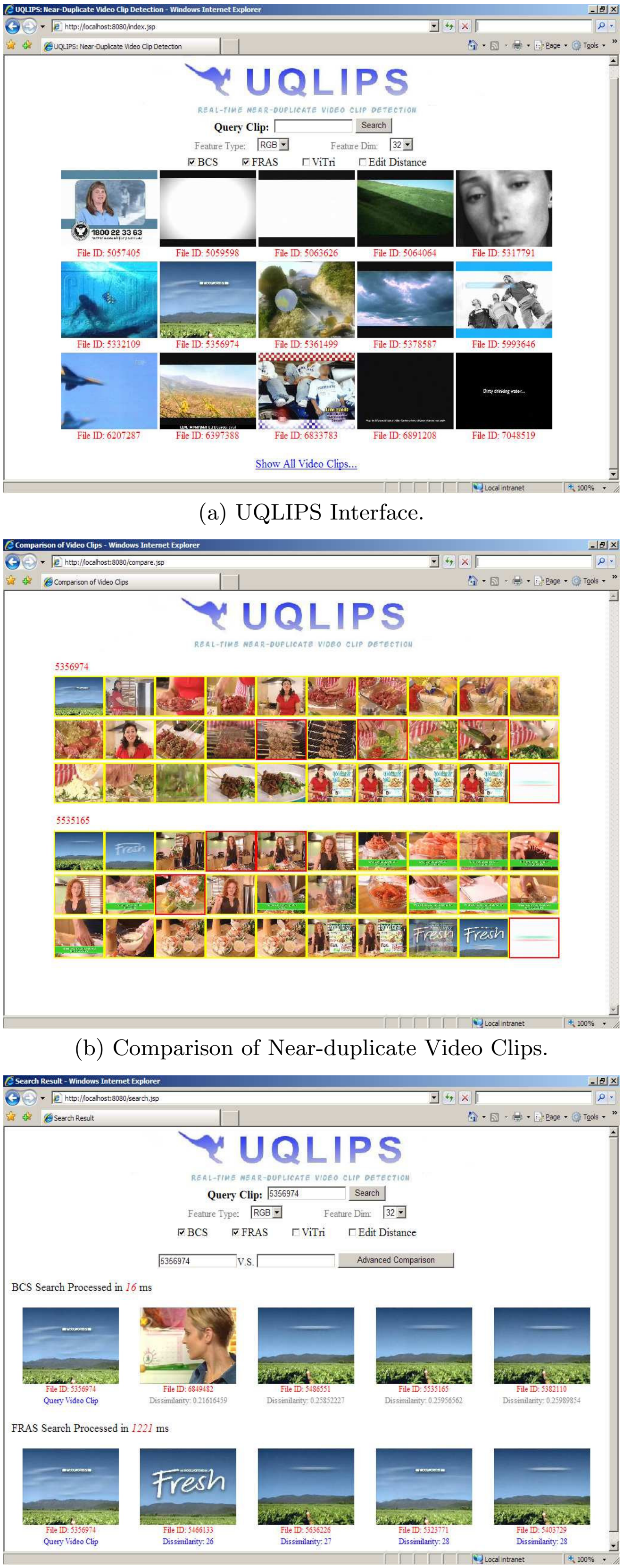
$$PED(S, S^{'}) = \begin{cases} \max(m, n) & m = 0 \; or \; n = 0 \\ \min\{PED(S_{m-1}, S^{'}_{n-1}) + p, & \\ PED(S_m, S^{'}_{n-1}) + 1, & \\ PED(S_{m-1}, S^{'}_n) + 1\} & otherwise \end{cases}$$

PED is different from the conventional edit distance in how to decide the similarity between two symbols. Given two frame symbols, $S[m-1] = S^{'}[n-1]$, their similarity is measured by a probability value computed by $p = \frac{|C - C^{'}| * |C^{'} - C|}{|C| * |C^{'}|}$, where C and $C^{'}$ are the clusters whose *ids* are $S[m-1]$ and $S[n-1]$, respectively. $|C - C^{'}|$ represents the number of video frames in C but not in $C^{'}$. Clearly, the time complexity of computing PED is quadratic.

3. DEMONSTRATION SCENARIO

This section shows the main functionalities of UQLIPS: NDVC search, comparing video clips, and search effectiveness and efficiency comparison of different methods.

Searching near-duplicate video clips: A user can submit a query clip by providing a sample clip, a video name or a URL, and specify his/her preference for search, e.g., se-



UQLIPS: Near-Duplicate Video Clip Detection - Windows Internet Explorer
http://localhost:8080/index.jsp
UQLIPS: Near-Duplicate Video Clip Detection
Page
Tools
UQLIPS
REAL-TIME NEAR-DUPLICATE VIDEO CLIP DETECTION
Search
Show All Video Clips...
Local intranet
100%

Query Clip:

Feature Type: RGB   Feature Dim: 32

☑ BCS ☑ FRAS □ ViTri □ Edit Distance

File ID: 5057405 File ID: 5059598 File ID: 5063626 File ID: 5064064 File ID: 5317791

File ID: 5332109 File ID: 5356974 File ID: 5361499 File ID: 5378587 File ID: 5993646

File ID: 6207287 File ID: 6397388 File ID: 6833783 File ID: 6891208 File ID: 7048519

(a) UQLIPS Interface.

Comparison of Video Clips - Windows Internet Explorer
http://localhost:8080/compare.jsp
Comparison of Video Clips
Page
Tools
UQLIPS
REAL-TIME NEAR-DUPLICATE VIDEO CLIP DETECTION
Local intranet
100%

5356974

5535165

(b) Comparison of Near-duplicate Video Clips.

Search Result - Windows Internet Explorer
http://localhost:8080/search.jsp
Search Result
Page
Tools
UQLIPS
REAL-TIME NEAR-DUPLICATE VIDEO CLIP DETECTION
Search
Advanced Comparison
Local intranet
100%

Query Clip: 5356974

Feature Type: RGB   Feature Dim: 32

☑ BCS ☑ FRAS □ ViTri □ Edit Distance

5356974 V.S.

BCS Search Processed in *16* ms

File ID: 5356974 Query Video Clip

File ID: 6849482 Dissimilarity: 0.21616459

File ID: 5486551 Dissimilarity: 0.25852227

File ID: 5535165 Dissimilarity: 0.25956562

File ID: 5382110 Dissimilarity: 0.25989854

FRAS Search Processed in *1221* ms

File ID: 5356974 Query Video Clip

File ID: 5466133 Dissimilarity: 26

File ID: 5636226 Dissimilarity: 27

File ID: 5323771 Dissimilarity: 28

File ID: 5403729 Dissimilarity: 28

(c) Comparison of Search Methods.

Figure 2: UQLIPS System Features.

lect the feature type (RGB or HSV color space), feature dimensionality (8, 16, 32, or 64) and tick the search methods (BCS, FRAS, ViTri or EDR) to be used. Figure 2(a) shows a simple interface of UQLIPS.

Comparing near-duplicate video clips: Among returned results, a user can select any two video clips to compare their detailed differences by browsing their key-frames at equal time intervals. Since the user may be interested in scanning the two videos simultaneously or finding the differences between two near-duplicates at particular positions, the system provides convenient ways to play the whole clips dynamically and compare the still key-frames at certain time stamps (see Figure 2(b)).

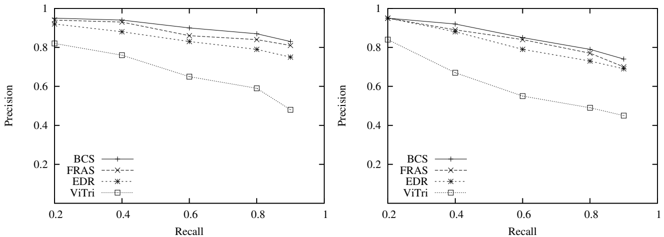
Comparing BCS, FRAS, ViTri and EDR: This system can provide the search results of up to four different methods together with their query response time. Figure 2(c) shows a snapshot of comparing the search results of BCS and FRAS. By using the functionality of *comparing near-duplicate video clips*, the user can easily identify which method returns more accurate results for the query. Each video clip is listed with its metadata (e.g., file name, video length, video format and file size, etc.) if required.

4. EXPERIMENTS

In this section, we report some experimental results of comparing four methods. We use the collection of more than 11,000 TV commercials with average length of about 60 seconds. Two popular feature spaces are used: RGB color space and HSV color space. Four feature datasets in 8-, 16-, 32- and 64-dimensionality for each color space were extracted for search purpose. Due to the space limit, we show the results in 64-dimensional space only. All the experiments were performed on Window XP platform with Intel Core 2 CPU (2.4 GHz) and 2.0 GB RAM. All the results reported are the average based on 20 query clips randomly selected from the database.

Figure 3 compares the precision-recall curves of four methods based on the ground-truth judged by human beings. As we can see, BCS, FRAS and EDR achieve comparable accuracy (e.g., the precision is greater than 80% when recall is 60%). BCS performs the best and ViTri performs the worst for both color features. This reveals that the moment of significance embedded in video content can be better described with the help of content changing trends, i.e., tendencies. BCS is able to identify the dominating content with its tendencies and measure the similarity along the tendencies. On the contrary, ViTri which estimates the video similarity based on the percentage of similar frames sometimes fails to resemble the relevance of videos according to human perception. FRAS and EDR are usually robust to strict temporal sequence matching, based on frame-to-frame similarity.

As for efficiency, the average search time for BCS is about 50 milliseconds in our system. However, the search time for FRAS and EDR is typically in seconds and minutes, respectively. This is because the time complexity of BCS similarity measure is linear. The time complexity of ViTri is quadratic in the number of representatives which is much smaller than the number video frames. The time complexity of both FRAS and EDR is quadratic in the number of video frames. However, FRAS is more efficient than EDR since inter-symbol matching is cheaper than inter-frame similarity computation.



(a) RGB dataset. (b) HSV dataset.

Figure 3: BCS vs. FRAS vs. EDR vs. ViTri.

In summary, it is clear that BCS is a practical solution for real-time near-duplicate video clip detection in large video databases. If temporal order is critical, our alternative approach, FRAS, can provide a more robust matching result.

5. CONCLUSIONS

UQLIPS is a prototype system that supports fast and robust NDVC search based on visual content. Given a query video clip, UQLIPS can quickly detect its near-duplicates, with an easy-to-use tool for users to see the differences. The effectiveness and efficiency of our core techniques, BCS and FRAS, can be tested at the demonstration.

6. REFERENCES

[1] YouTube. http://www.youtube.com.

[2] D. A. Adjeroh, M.-C. Lee, and I. King. A distance measure for video sequences. *Computer Vision and Image Understanding*, 75(1-2):25–45, 1999.

[3] H. S. Chang, S. Sull, and S. U. Lee. Efficient video indexing scheme for content-based retrieval. *IEEE Trans. Circuits Syst. Video Techn.*, 9(8):1269–1279, 1999.

[4] L. Chen, M. T. Özsu, and V. Oria. Robust and fast similarity search for moving object trajectories. In *SIGMOD Conference*, pages 491–502, 2005.

[5] S.-C. S. Cheung and A. Zakhor. Efficient video similarity measurement with video signature. *IEEE Trans. Circuits Syst. Video Techn.*, 13(1):59–74, 2003.

[6] Y. Ke, R. Sukthankar, and L. Huston. Efficient near-duplicate detection and sub-image retrieval. In *ACM Multimedia*, pages 869–876, 2004.

[7] H. T. Shen, B. C. Ooi, X. Zhou, and Z. Huang. Towards effective indexing for very large video sequence database. In *SIGMOD Conference*, pages 730–741, 2005.

[8] D. Zhang and S.-F. Chang. Detecting image near-duplicate by stochastic attributed relational graph matching with learning. In *ACM Multimedia*, pages 877–884, 2004.

[9] X. Zhou, X. Zhou, and H. T. Shen. Efficient similarity search by summarization in large video database. In *ADC*, pages 161–167, 2007.

[10] X. Zhu, A. K. Elmagarmid, X. Xue, L. Wu, and A. C. Catlin. Insightvideo: toward hierarchical video content organization for efficient browsing, summarization and retrieval. *IEEE Transactions on Multimedia*, 7(4):648–666, 2005.