Challenges and Techniques for Effective and Efficient Similarity Search in Large Video Databases

Jie Shao
supervised by Heng Tao Shen and Xiaofang Zhou
School of Information Technology & Electrical Engineering
The University of Queensland
Brisbane QLD 4072, Australia
{jshao, shenht, zxf}@itee.uq.edu.au

ABSTRACT

Searching relevant visual information based on content features in large databases is an interesting and changeling topic that has drawn lots of attention from both the research community and industry. This paper gives an overview of our investigations on effective and efficient video similarity search. We briefly introduce some novel techniques developed for two specific tasks studied in this PhD project: video retrieval in a large collection of segmented video clips, and video subsequence identification from a long unsegmented stream. The proposed methods for processing these two types of similarity queries have shown encouraging performance and are being incorporated into our prototype system of video search named UQLIPS, which has demonstrated some marketing potentials for commercialisation.

1. INTRODUCTION

Recently, multimedia search continuously attracts increasing interest from many researchers as well as commercial organizations. Among the media types, video carries the richest content in daily information communication and acquisition. With the advances of hardware (e.g., the plummeting of storage cost) and software (e.g., the popularity of video editing utility), we are experiencing tremendous amount of video data today in many fields such as in personal, commercial and organizational video archives. In addition, with the wide spread of broadband access, videos become very popular on the Web. According to comScore [1], a leader in measuring the digital world, nearly 134 million Americans viewed more than 9 billion online videos in July 2007 alone. Online viewers watched an average of 181 minutes video during the month, nearly 30 more than January 2007. The average online video viewer consumed 68 videos (or more than 2 videos per day), and the number is expected to keep growing. Clearly, the rapid increase in the generation and dissemination of digital videos in both centralized video archives and distributed video resources on the Web has created an urgent need for video search engines to facilitate retrieving relevant information of interest.

PVLDB '08, August 23-28, 2008, Auckland, New Zealand
Copyright 2008 VLDB Endowment, ACM 978-1-60558-306-8/08/08

Generally, a video can be viewed as a multi-modality of visual, audio, textual and motion features [28]. Currently available commercial video search engines such as Google Video only provide searches based on metadata or surrounding textual information and do not exploit the content information in a nature and intuitive way. In some online video sharing sites such as YouTube, video search is converted into a query of the specified keyword(s) in annotation data provided by users. Although this approach is directly related to video semantics, in general it is unsuitable for similarity-based video search in large repositories, since manually annotating involves too many human efforts and is a tedious, cumbersome and time-consuming task. Moreover, text descriptors can often be biased and incomplete, since annotations are intrinsically subjective and only represent partial information. Different from text retrieval that typically involves keyword query, video search is actually more complicated due to its complex information embedded. In this research, we focus on the video modality of a sequence of frames, each of which is typically represented by some low-level feature [32] which is referred to as *video content*, such as color distribution, texture pattern or shape structure. In other words, we exploit the inherent visual information for content-based video search.

This research is initially motivated by the TV commercial detection project of our collaborator from ACNielsen, a well-known media research organization. The intriguing application is automatic recognition of TV commercials, which is an essential step in TV broadcast monitoring. When a company contracts several TV stations for certain commercials, it often asks a marketing survey company to track whether its commercials are actually broadcasted as contracted (when - before/after/during certain popular programs, and how - the exact durations and times, etc.). Some companies may also approach such marketing survey companies to seek information about how their competitors market their products. While a commercial is given to some different TV stations for broadcasting, it can be aired with a number of variations, such as station-specific parameters (e.g., frame rate, aspect ratio and resolution), and inserts of different local contact information or products (e.g., a supermarket would like to insert different products on sale in its TV commercial 'template'). Thus, the various versions of TV commercials broadcasted by different TV stations at different times are 'similar' but not exactly 'same'.

Video similarity search also has other practical applications. A typical one is enforcing copyright compliance. Video content owners would like to be aware of any use of their material, in any media or representation. Consider, for example, the producers of certain movie scenes may want to identify whether/where their original works have been re-used by others, even with some kind of re-

mixing for multimedia authoring. Since videos are easy to copy, reformat, modify and republish, duplicate or near-duplicate videos often spread in the absence of central management or monitoring, thus there are many such videos distributed on the Web. It is reported that YouTube has recently encountered some legal issues related to copyright infringement, because copyrighted videos explicitly requested to be banned by their copyright owners are often found to be uploaded again by other users with some minor changes. Therefore, an important problem now faced by these video sharing sites is how to automatically perform accurate and fast similarity search for an incoming video clip against its huge database, to avoid copyright violation. Meanwhile, since the retrieval efficiency will be hampered if a large number of search results are essentially almost-identical, database purge also contributes to high-quality ranking for video search results [34].

In content-based search, each video is transformed to feature vectors, which are points in a high-dimensional space. Matching of similar videos is often translated into searches among these feature vectors [14, 18, 20, 16, 35, 8, 25]. The number of feature vectors depends on the length of video. Due to the high complexity of video features, scanning on all these vectors results in high computational cost and is strongly undesirable. Novel indexing and query processing techniques are indispensable to manipulate large video databases. We identify three essential issues that have to be addressed in managing such a system as follows: first, obtaining an effective and compact video representation; second, measuring the similarity of videos that ideally can match the 'similar' notion of human perception; third, organizing the compact representations with an indexing structure, together with an efficient search strategy.

In this PhD project, we investigate two different aspects of content-based video similarity search:

- *Video clip retrieval*, which conventionally returns similar clips from a large collection of videos which have been either chopped up into similar lengths or cut at content boundaries.
- *Video subsequence identification*, which aims at finding if there exists any subsequence of a long database video that shares similar content to a query.

The primary difference between these two scenarios of similarity queries is that, for the former, the clips for search have already been segmented and are always ready for similarity ranking [35, 8, 25], the latter is a typical subsequence matching problem [14, 18, 20, 16] (which is conceptually analogous to subsequence matching in time series [10]). Because the boundary, and even the length of target subsequence are not available at beginning, choosing which subsequences for evaluating similarities is not pre-known.

Our objective is to find a generic database management solution towards effectively and efficiently searching similar videos, with tolerance to different variations introduced during not only transformation process but post-production editing. Existing related studies are mainly focused on a more specific problem of *co-derivative* video detecting. In particular, we are interested in searching visually relevant video, even if there exists some transformation distortion, partial content re-ordering, insertion, deletion, or replacement. This scenario introduces additional complexity. To date, we have worked out a series of new methodologies to tackle the aforementioned rising challenges to render video similarity search more practical, in terms of both speed and accuracy. In summary, we make the following technical contributions:

- We develop a batch k Nearest Neighbor (kNN) search algorithm [23, 24] which efficiently processes a bunch of individual kNN searches on the same database simultaneously to significantly reduce the computational overhead of content-based video search systems, without compromising the accuracy of results.
- On top of this batch query processing strategy, we propose a graph transformation and matching approach to video subsequence identification, with extension to identify the occurrence of potentially different ordering or length due to content editing for effective identification.
- We propose a novel video clip representation model called Bounded Coordinate System (BCS) [26], which statistically summarizes a video clip into a single representative by analyzing the correlation of frame content existing in feature space, to facilitate fast retrieval in large video clip databases. This method goes beyond a straightforward adaptation of our early proposal Video Triplet (ViTri) [25].
- Moreover, we further consider a more robust methodology of similarity assessment which evaluates the probabilities of vectorial distribution consistency. Compared with summarizing the correlation of content features for comparison, directly exploiting the criterion of distributional discrepancy is a more reliable and general solution for video clip retrieval.

This paper aims at embedding our past and ongoing research efforts in the context of a whole PhD project. We intend to show some threads for relating the four pieces of work, and put emphasis on discussing the motivation of each individual work. Since some results are not published yet, to be self-contained, we also briefly explain the main idea of each technique here. The rest of the paper is organized as follows. We first give some background information in Section 2. The framework of our proposals is given in Section 3. Section 4 briefly introduces our prototype system UQLIPS which is currently under further development. Finally, we conclude and mention some scheduled extensions of this research as future objectives in Section 5.

2. BACKGROUND

In conventional content-based similarity search such as image retrieval, a query consumes a single kNN search. High-dimensional indexing has been extensively studied in the database literature [4]. Although tree-based index structures work well in low to medium dimensional spaces, a simple sequential scan usually performs better at higher dimensionality [33]. To tackle the notorious 'curse of dimensionality', substantial progresses have been made, which can be generally classified into five approaches: tree-like structure such as X-tree [2], data compression such as VA-file [33], dimensionality reduction and hybrid of tree-like structure such as LDR [5], transformation to one-dimension such as iDistance [15] and approximate search such as LSH [12]. These techniques are all concerned with facilitating single kNN search. However, a distinguishing characteristic of video search is that, each video is described by a sequence of feature vectors, so as to the query. Denote a query clip as $Q = \{q_1, q_2, \ldots, q_m\}$ and a database video as $P = \{p_1, p_2, \ldots, p_n\}$ (i.e., Q and P have m and n feature vectors respectively), to determine whether P is similar to Q or contains Q, typically for each $q_i \in Q$, a search is first performed in P to retrieve the similar feature vectors to q_i. After completing all the kNN searches, an overall similarity is then computed, i.e., a single content-based video search usually involves m individual kNN searches. This bottleneck is restricting most existing content-based video search systems to test on relatively small video data sets. Unfortunately, there are few studies addressing how to

Figure 1: Visually similar videos, but not copies.

efficiently process a batch of kNN simultaneously. All-nearest-neighbors queries [36] investigates how to schedule a number of individual kNN searches by the spatial proximity to maximize the buffer hit ratio in the context of two-dimensional spatial databases. It is proposed to order the queries with a space filling curve, e.g., Hilbert curve, to maximize the point locality. However, in high-dimensional space for multimedia applications, this approach becomes no longer applicable, since the data set is presumed to be either indexed with R-tree or spatial hashing. Therefore, it does not scale well with dimensionality.

In multimedia, most existing research efforts are on content-based video copy detection [14, 18, 20, 16, 35], which is regarded as a complementary mechanism of video watermarking. Their primary focus is to detect any video sharing exactly the same original source with query, but can be altered with some global transformation, such as different formats, different resolutions, changing brightness, changing contrast, changing saturation, cropping, overlaying a logo, etc. Our research addresses a more changeling problem of searching *visually similar* videos. Different from copy detection which normally considers transformation distortions only, a visually similar video can be further relaxed to be changed with content editing at frame or shot level (swap, insertion, deletion, or substitution), thus could have different ordering or length with original source. For example, Figure 1 shows two TV commercials for Tourism New South Wales, Australia. Each of them is displayed with 5 sampled frames extracted at the same time stamps. They are highly similar, but not copies. Another example is the extended cinema version of Toyota commercial and its shorter TV version, which obviously are not copies of each other by definition.

To further search videos with changes from query due to content editing, a number of algorithms have been proposed to evaluate video similarity. In case of retrieving relevant videos from a collection of well segmented clips, some summarization techniques can be applied to obtain compact video representations. Video similarity then can be estimated based on these compact representations. Two typical examples are ViSig [8] and ViTri [25] that both estimate the percentage of visually similar frames. In [8], a randomized algorithm is proposed to select a number of seed frames and assigns a small collection of closest frames called Video Signatures to the set of seed frames. However, depending on the relative positions of the seed frames and ViSigs, this randomized algorithm may sample non-similar frames from almost-identical videos. In [25], each video is summarized into a set of clusters, each of which is modelled as a hyper-sphere called Video Triplet described by its position, radius, and density. Each video is represented by a much smaller number of hyper-spheres. Video similarity is then approximated by the total volume of intersections between two hyper-spheres multiplying the smaller density of clusters.

In case of identifying relevant subsequence, these methods of video retrieval become inapplicable. Video subsequence matching techniques using a fixed length sliding window at every possible position of database sequence for exhaustive comparison [14, 18, 20] are *not efficient*, especially for seeking over a long-running video. Although a temporal skip scheme using similarity upper bound [16, 35] can accelerate search process by reducing the number of candidate subsequences, under the scenario that actually a target subsequence could have different ordering or length with a query, these methods suffer from being *not effective*.

Since the temporal characteristic naturally models a video sequence as a trajectory in vector space, various time series similarity measures can be considered, such as Mean distance (normalized pairwise distance) [19], Dynamic Time Warping (DTW) [17], Longest Common Subsequence (LCSS) [30], and Edit distance (e.g., Edit distance with Real Penalty (ERP) [7]), all of which can be extended to measure the similarity of multi-dimensional trajectories and applied for video matching. For instance, Mean distance is adopted in [16], DTW is adopted in [9], LCSS is adopted in [6], and Edit distance is adopted in [3]. However, for the specific problem of measuring video similarity, when dealing with temporal order, frame alignment, gap and noise together, all these similarity measures are insufficient in some aspect.

3. OUR PROPOSALS

In this section, we first introduce our batch kNN search enabled video subsequence identification technique, then discuss two proposals for video clip retrieval.

3.1 Strategies for Subsequence Identification

We develop an effective and efficient strategy for temporal localization of similar content from a long unsegmented video stream, with particular consideration that target subsequence may be approximate occurrence of potentially different ordering or length with respect to query.

3.1.1 Batch Search for Similar Frame Retrieval

As mentioned above, a single content-based video search usually involves a number of individual kNN searches. Our first work is motivated by the fact that traditional indexing methods do not provide adequate efficiency for this process. Since normally nearby feature vectors in a video are similar, a series of separate kNN searches will incur lots of expensive I/O cost for random disk accesses and CPU cost for distance computations, which crucially affect the overall query performance. Intuitively, some results of next query could be probably contained in the results of previous queries. We define batch k Nearest Neighbor search as a batch operation that performs a bunch of individual kNN searches on the same database simultaneously. Consequently, the speed of similar frame retrieval, which serves as the first step in our query processing strategy for video subsequence identification described next, can be improved significantly.

Effectively utilizing the overlaps among queries to the maximal extent is nontrivial. Optimizations of multiple queries have been well studied in the context of relational databases [22]. The proposed techniques include elimination of common sub-expressions, re-ordering of query plans, using materialized views, pre-fetching and caching of input data values, etc. However, in high-dimensional feature data sets for multimedia applications, the technique used for minimizing the number of accesses to database points is different from that of minimizing the number of accesses to database relations. [23] is the first work that identifies pruning condition tightening, the opportunity to further reduce the total number of candidates through batch operation. Observing the overlapped candidates (or search space) of a pervious query may help to further reduce the candidate sets of subsequent queries, we propose Dynamic Query Ordering (DQO) execution for efficiently processing batch kNN

search in high-dimensional space, with advanced optimizations of both I/O cost and CPU cost. DQO can progressively find a query order such that the results of previously processed queries can be maximally re-used by next query. In additional to the experiment results reported in [23] where successive video frames are used for the query video clip, [24] further demonstrates the significance of our batch kNN query processing strategy when sampled frames are used.

3.1.2 Graph Transformation and Matching

Temporal order is an essential characteristic of video sequences. Following their temporal orders, a submitted query Q and long database video P can be placed along two one-dimensional temporal lines. This motivates us to investigate the mapping relationship between Q and P by a bipartite graph.

The main steps of our query processing can be described as follows. With the proposed batch query algorithm to retrieve similar frames, the mapping relationship between Q and P is first represented by a bipartite graph. This is a preliminary step analogous to building element-to-element correspondence in time series matching. Then, the densely matched parts along the long sequence P are extracted. In this way, a large portion of irrelevant parts are safely pruned and we can locate all the possibly similar video subsequences as promising candidates for similarity evaluations. Next, to effectively but still efficiently identify the actually similar subsequence, a novel filter-and-refine strategy is proposed to prune some irrelevant subsequences. During the filtering stage, imposing a one-to-one mapping constraint similar in spirit to that of [21], Maximum Size Matching (MSM) [27] is deployed for each subgraph constructed by the query and candidate subsequence to rapidly filter some actually non-similar subsequences with lower computational cost. During the refinement stage, Sub-Maximum Similarity Matching (SMSM) is devised to identify the subsequence with the highest aggregate score from all candidates with relatively higher computational cost, according to a robust video similarity model which incorporates visual content, temporal order, frame alignment information for accurate identification. In a nutshell, by exploiting the mapping relationship between Q and P, we transform video subsequence identification to a matching problem in a bipartite graph for processing variable length comparison over database video P with query Q.

Compared with existing work, our strategy has two distinctive features. First, in contrast to the fast sequential search scheme applying *temporal pruning* to accelerate search process [16, 35] which assumes query and target subsequence are strictly of the same ordering and length, it adopts *spatial pruning* to avoid seeking over the entire database sequence of feature vectors for exhaustive comparison. Second, it does not involve video segmentation required by the proposals based on shot boundary detection such as [6, 21]. Shot resolution, which could be a few seconds in duration, is usually too coarse to accurately locate a subsequence boundary. Meanwhile, our strategy based on frame sub-sampling is capable of identifying video content containing ambiguous shot boundaries (such as dynamic commercial, TV program lead-in and lead-out subsequences).

3.2 Strategies for Clip Retrieval

In video clip retrieval, each database video is well segmented and available for similarity evaluation. It can be viewed as $X = \{x_1, x_2, \ldots, x_n\}$, where each element $x_i \in \mathbb{R}^d$ is a d-dimensional feature vector point representing a frame in X, and n is the number of sampling frames. It is often unnecessary to maintain the full fidelity of feature vector representations so effective and compact video representations can be applied to alleviate high computational complexity. In general, there are two types of summarization techniques for measuring similarity: higher order techniques which summarize all feature vectors of a video clip into some high-level descriptions, such as ViTris, and first-order techniques which summarize a video clip by a small set of representative feature vectors, such as ViSigs. Then, the similarity between two video clips can be estimated by comparing their corresponding compact representations.

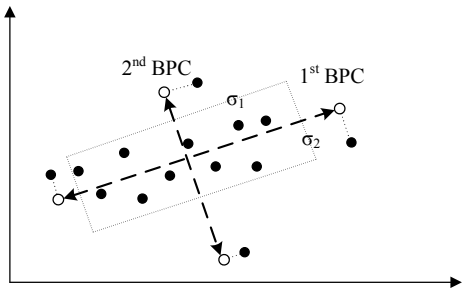
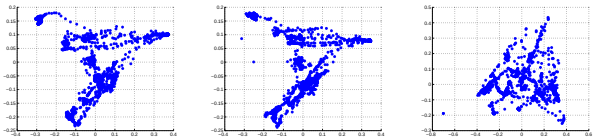


Figure 2: Bounded Principle Components.

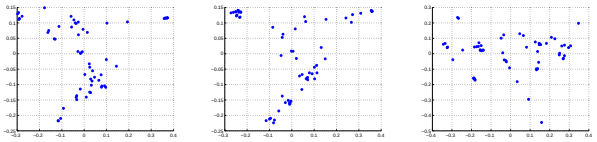
3.2.1 Statistical Summarization of Content Features

We first review Bounded Coordinate System (BCS) [26], which is the first single representative that globally captures the dominating content and content changing trends of a video clip by exploiting the tendencies of content features. As illustrated with a simple example in Figure 2, the dots represent the frame features of a video clip. Principal Component Analysis (PCA) can project the data points to a new coordinate system such that the greatest variance comes to lie on the first principal component, the second greatest variance on the second principal component, and so on. Conventionally, principal components only indicate the directions of coordinate axes. Here we adopt a bounded scheme called Bounded Principle Component (BPC). For a principal component Φ_i identifying a direction, its corresponding BPC $\ddot{\Phi}_i$ identifies a line segment bounded by two furthermost projections on Φ_i, as shown by two circles in Figure 2, with the length $||\ddot{\Phi}_i||$. BPCs indicate the ranges of feature vector scattering along certain orientations. Given a video clip X, its compact representation $BCS(X) = (O, \ddot{\Phi}_1, \ldots, \ddot{\Phi}_d)$ is the mean (origin) for all x_i denoted as O, and d BPCs (orientations and ranges). Independent of frame number n, BCS only records a centering d-dimensional point and d BPCs to represent a video clip. A BCS actually consists of $(d+1)$ d-dimensional vectors. Since real video data often have some noise points, to eliminate this effect and capture data distribution more robustly, the length of BPC can be re-defined with standard deviation, which measures the statistical dispersion of data projections. A BPC $\ddot{\Phi}_i$ identifies a line segment bounded by σ_i, where σ_i is the standard deviation indicating the average distance of all data points from a projection on Φ_i to the origin of coordinate system. Its length $||\ddot{\Phi}_i|| = 2\sigma_i$. The dashed rectangle in Figure 2 shows two corresponding BPCs by σ_1 and σ_2. The similarity measure of videos is transformed into a comparison of the corresponding BCSs. Given video clips X and Y with $BCS(X) = (O^X, \ddot{\Phi}_1^X, \ldots, \ddot{\Phi}_d^X)$ and $BCS(Y) = (O^Y, \ddot{\Phi}_1^Y, \ldots, \ddot{\Phi}_d^Y)$, their similarity can be intuitively estimated by performing translation, rotation and scaling operations to match two BCSs. The distance of two BCSs can be computed by $||O^X - O^Y|| + \sum_{i=1}^{d} ||\ddot{\Phi}_i^X - \ddot{\Phi}_i^Y||$.

The compact video representation together with its linear time



(a) Similar video clips A and B, and a dissimilar video clip C.



(b) Simple random sampling, select 60 elements.

Figure 3: Vectorial distributions in feature space.

similarity measure makes real-time search from large video clip collections feasible. We can further use the optimal one-dimensional transformation accommodated with B^+-tree [25] for BCS indexing. Given the understanding that a video clip often shows a moment of significance, from human perception, different dominating visual content of clips may express different significance. At the same time, different content changes may also suggest different meanings. In practice, BCS based on correlation analysis usually works quite well. It is observed from large-scale experiments that BCS achieves very high search accuracy, and can convey certain degree of semantic information.

3.2.2 Distribution-based Similarity Measures

Same with other higher order summarization techniques, theoretically BCS has a limitation that it assumes a restricted form of density model. Specifically, accompanied with PCA, it implicitly brings an assumption that the data points in each video clip are drawn from a Gaussian normal distribution, which is statistically unwarranted in real world. PCA can be ill-adapted to the sets of data points with intrinsic nonlinear correlations, which possibly compromises the effectiveness of similarity search based on BCS. For more robust search the enormous amount of and extremely diverse video content on the Web, we address this problem by modelling distribution-based similarity measure for video clip retrieval.

We propose a novel collective perspective of exploiting the distributional discrepancy of samples for assessing the similarity between two ensembles of points. Several ideas of non-parametric hypothesis tests in the literature of multivariate statistics are utilized to check the hypothesis whether two ensembles of points are from a same distribution. The two specific methods adopted are Friedman-Rafsky (FR) test [11], and a more efficient Maximum Mean Discrepancy (MMD) test [13]. FR test is a multivariate generalization of classical Wald-Wolfowitz (WW) test [31], and MMD test is a kernel method which employs the unit balls in a universal Reproducing Kernel Hilbert Space (RKHS) [29] as its function class. The test statistic of either method can be expressed as the probability that two ensembles of points are consistent with a same distribution, which itself is not explicitly known. Figure 3(a) exemplifies the vectorial distributions of three video clips in feature space. For visualization, the original 32-dimensional feature vectors are projected with the first two PCA coefficients. Each point in the plots stands for a frame. Actually video clips A and B are similar ones, while video clip C is irrelevant. It becomes evident that the distributions of similar videos are with much likeness, while dissimilar one is quite different. Figure 3(b) plots the results of

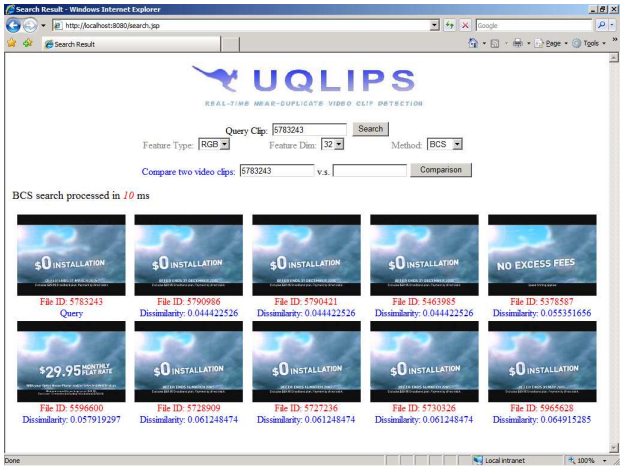


Figure 4: Interface of UQLIPS.

a *simple random sampling* technique (without replacement). Experimentally, we observe that satisfactory retrieval results can be achieved even when small to moderate sample size of representative points are used, thus the computational cost of distribution-based methods can be alleviated significantly.

The main advantage of this proposal is that, no prior knowledge about the underlying data distributions of the point sets under study has to be assumed. It provides a more comprehensive analysis that captures the essence of invariant distribution information for retrieving video clips. In fact, the design philosophy of BCS resorting to content correlation can be regarded as some attempts to roughly reflect the statistical distributions of data points by some coarse content tendency. The new proposal based on the criterion of distributional discrepancy is more direct, reliable (more descriptive local information will be exploited) and general (can not only fit a particular parametric form), which shows better retrieval quality in our preliminary experiments.

4. PROTOTYPE SYSTEM

We show in Figure 4 a snapshot of a Web-based video search system named UQLIPS developed at the University of Queensland. Currently it has a large database of more than 50,000 video clips, where BCS can retrieval similar video clips with very satisfactory search accuracy in milliseconds. This system has been demonstrated at some international conferences and to industry people. Its marketing opportunity is highly commended by UniQuest (the Australia's largest and most successful university commercialisation group), and expected to be further invested in different ways. More functionalities such as the module of video subsequence identification are currently being incorporated into UQLIPS to make this prototype system more practical and powerful.

5. CONCLUSIONS AND PERSPECTIVES

The rapid advances in multimedia and network technologies make video search become a key part of the future of digital media. This paper discusses the challenges of similarity search in large video databases, and gives an overview of our proposed techniques for this problem. Our work extends the investigations of video copy detection not only in the aspect of potentially different length but also allowing flexible temporal order (tolerance to partial re-ordering). Currently we assume the long video sequence for subsequence matching is pre-stored in database. We are developing an accurate and

fast system for online detection over continuous video streams. This module will also be part of our system UQLIPS. In additional, we will work on search result clustering based on both text and content information, to fully utilize multi-modality video features.

6. ACKNOWLEDGMENTS

The work reported in this paper is partially supported by an Australia Research Council Grant DP0663272 and a UniQuest Pathfinder Grant. The author would like to thank Zi Huang, Yijun Li, Liping Wang and Xiangmin Zhou for many research collaborations.

7. REFERENCES

[1] http://www.comscore.com/press/release.asp?press=1678.

[2] S. Berchtold, D. A. Keim, and H.-P. Kriegel. The x-tree : An index structure for high-dimensional data. In *VLDB*, pages 28–39, 1996.

[3] M. Bertini, A. D. Bimbo, and W. Nunziati. Video clip matching using mpeg-7 descriptors and edit distance. In *CIVR*, pages 133–142, 2006.

[4] C. Böhm, S. Berchtold, and D. A. Keim. Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Comput. Surv.*, 33(3):322–373, 2001.

[5] K. Chakrabarti and S. Mehrotra. Local dimensionality reduction: A new approach to indexing high dimensional spaces. In *VLDB*, pages 89–100, 2000.

[6] L. Chen and T.-S. Chua. A match and tiling approach to content-based video retrieval. In *ICME*, pages 417–420, 2001.

[7] L. Chen and R. T. Ng. On the marriage of lp-norms and edit distance. In *VLDB*, pages 792–803, 2004.

[8] 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.

[9] C.-Y. Chiu, C.-H. Li, H.-A. Wang, C.-S. Chen, and L.-F. Chien. A time warping based approach for video copy detection. In *ICPR (3)*, pages 228–231, 2006.

[10] C. Faloutsos, M. Ranganathan, and Y. Manolopoulos. Fast subsequence matching in time-series databases. In *SIGMOD Conference*, pages 419–429, 1994.

[11] J. H. Friedman and L. C. Rafsky. Multivariate generalizations of the wald-wolfowitz and smirnov two-sample tests. *Ann. Statist.*, 7(4):697–717, 1979.

[12] A. Gionis, P. Indyk, and R. Motwani. Similarity search in high dimensions via hashing. In *VLDB*, pages 518–529, 1999.

[13] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. J. Smola. A kernel method for the two-sample-problem. In *NIPS*, pages 513–520, 2006.

[14] A. Hampapur, K.-H. Hyun, and R. M. Bolle. Comparison of sequence matching techniques for video copy detection. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 194–201, 2002.

[15] H. V. Jagadish, B. C. Ooi, K.-L. Tan, C. Yu, and R. Zhang. idistance: An adaptive b^+-tree based indexing method for nearest neighbor search. *ACM Trans. Database Syst.*, 30(2):364–397, 2005.

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

[17] E. J. Keogh. Exact indexing of dynamic time warping. In *VLDB*, pages 406–417, 2002.

[18] C. Kim and B. Vasudev. Spatiotemporal sequence matching for efficient video copy detection. *IEEE Trans. Circuits Syst. Video Techn.*, 15(1):127–132, 2005.

[19] S.-L. Lee, S.-J. Chun, D.-H. Kim, J.-H. Lee, and C.-W. Chung. Similarity search for multidimensional data sequences. In *ICDE*, pages 599–608, 2000.

[20] M. R. Naphade, M. M. Yeung, and B.-L. Yeo. A novel scheme for fast and efficient video sequence matching using compact signatures. In *Storage and Retrieval for Image and Video Databases (SPIE)*, pages 564–572, 2000.

[21] Y. Peng and C.-W. Ngo. Clip-based similarity measure for query-dependent clip retrieval and video summarization. *IEEE Trans. Circuits Syst. Video Techn.*, 16(5):612–627, 2006.

[22] T. K. Sellis. Multiple-query optimization. *ACM Trans. Database Syst.*, 13(1):23–52, 1988.

[23] J. Shao, Z. Huang, H. T. Shen, X. Zhou, and Y. Li. Dynamic batch nearest neighbor search in video retrieval. In *ICDE*, pages 1395–1399, 2007.

[24] J. Shao, Z. Huang, H. T. Shen, X. Zhou, E.-P. Lim, and Y. Li. Batch nearest neighbor search for video retrieval. *IEEE Transactions on Multimedia*, 10(3):409–420, 2008.

[25] 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.

[26] H. T. Shen, X. Zhou, Z. Huang, J. Shao, and X. Zhou. Uqlips: A real-time near-duplicate video clip detection system. In *VLDB*, pages 1374–1377, 2007.

[27] D. R. Shier. Matchings and assignments. In J. L. Gross and J. Yellen, editors, *Handbook of Graph Theory*, pages 1103–1116. CRC Press, 2004.

[28] C. Snoek and M. Worring. Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools Appl.*, 25(1):5–35, 2005.

[29] I. Steinwart. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93, 2001.

[30] M. Vlachos, D. Gunopulos, and G. Kollios. Discovering similar multidimensional trajectories. In *ICDE*, pages 673–684, 2002.

[31] A. Wald and J. Wolfowitz. On a test whether two samples are from the same population. *Ann. Math. Statist.*, 11(2):147–162, 1940.

[32] H. Wang, A. Divakaran, A. Vetro, S.-F. Chang, and H. Sun. Survey of compressed-domain features used in audio-visual indexing and analysis. *J. Visual Communication and Image Representation*, 14(2):150–183, 2003.

[33] R. Weber, H.-J. Schek, and S. Blott. A quantitative analysis and performance study for similarity-search methods in high-dimensional spaces. In *VLDB*, pages 194–205, 1998.

[34] X. Wu, A. G. Hauptmann, and C.-W. Ngo. Practical elimination of near-duplicates from web video search. In *ACM Multimedia*, pages 218–227, 2007.

[35] J. Yuan, L.-Y. Duan, Q. Tian, S. Ranganath, and C. Xu. Fast and robust short video clip search for copy detection. In *PCM (2)*, pages 479–488, 2004.

[36] J. Zhang, N. Mamoulis, D. Papadias, and Y. Tao. All-nearest-neighbors queries in spatial databases. In *SSDBM*, pages 297–306, 2004.