Fast and Robust Short Video Clip Search Using an Index Structure

Junsong Yuan[1,2], Ling-Yu Duan[1], Qi Tian[1], Changsheng Xu[1]
[1] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
[2] Dept. of ECE, National University of Singapore
{jyuan, lingyu, tian, xucs}@ i2r.a-star.edu.sg

ABSTRACT
In this paper, we present an index structure-based method to fast and robustly search short video clips in large video collections. First we temporally segment a given long video stream into overlapped matching windows, then map extracted features from the windows into points in a high dimensional feature space, and construct index structures for these feature points for querying process. Different from linear-scan similarity matching methods, querying process can be accelerated by spatial pruning brought by an index structure. A multi-resolution kd-tree (mrkd-tree) is employed to complete *exact K-NN Query* and *range query* with the aim of fast and precisely searching out all short video segments having the same contents as the query. In terms of feature representation, rather than selecting representative key frames, we develop a set of spatial-temporal features in order to globally capture the pattern of a short video clip (e.g. a commercial clip, a lead in/out clip) and combine it with the color range feature to form video signatures. Our experiments have shown the efficiency and effectiveness of the proposed method that the very first instance of a given 10-sec query clip can be identified from a 10.5-hour video collection in tens of milliseconds. The proposed method has been also compared with the fast sequential search algorithm.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval–*Search process.* I.4.7 [**Image Processing and Computer Vision**]: Feature Measurement–*Feature representation*

General Terms
Algorithms, Design, Experimentation

Keywords
Video Similarity Search, Video Content Identification, Fast Query, Spatial-Temporal Feature

MIR'04, October 15–16, 2004, New York, New York, USA.

1. INTRODUCTION
Content based video search and retrieval has attracted high research interests. According to different user requirements, we may have different kinds of video search intentions including the detection of duplicate/near-duplicate entry of a given clip, the identification of recurrent instances of a given clip, and similar video content search in a sense of coarse or clear semantic meanings. Application comprises video copyright management [6] [13], video content identification [1] [6] [9] [12] [15] and content-based video retrieval [2] [3] [4] [14].

For content-based video retrieval and identification, most existing search methods are derived from the sequential correlation matching widely used in the signal processing domain. Those sequential matching methods usually focus much more on feature extraction and a similarity matching function in order to search desired video content. The search efficiency is ignored to some extent. Although there exist some techniques to improve the linear scanning speed, such as temporal pruning [1] or coarser granularity searching [12], their searching time-complexity still remains at least linear to the size of database as it usually requires exhaustively seeking out the whole data. In this paper, we present a fast and robust short video clip searching scheme using an index structure. This is motivated by the flexible and efficient query capabilities brought by fruitful research in high-dimensional index structures [5]. We will employ an mrkd-tree index structure [10] and apply different query strategies to short video clip search in this paper. A comparison will be performed between an index-based work and the temporal pruning-based active searching [1].

From the database point of view, video search can be treated as a querying problem [5] [7] [8]. That is, the essential purpose of video search is to find out those contents that are the most similar to the query, namely the "neighbors" of a query point in the feature space. Different query strategies can be applied to fulfill different applications [5]. For example, *K-NN Query* can retrieve nearby points by their similarity ranking and then it is suitable for content based retrieval task. In contrast, *Range Query* is able to find all individual points within a certain distance from a query and then it can be used for duplicate/near-duplicate detection or content identification tasks. Such query strategies have ever been successfully applied for fast content-based image retrieval. Nevertheless, different from an image database, a video database contains large amounts of image sequences exhibiting a temporal order and a high redundancy. Indexing each image frame is thus ineffective and practically impossible. In our approach, we chop

the long video stream into a series of overlapped video segments of a fixed size (we call them as "3D volume elements"), and build an index structure for these volume elements instead of individual frames. Compared with traditional key-frames based representation, our approach can avoid the shot segmentation and key frames selection. This is useful for identifying video content containing ambiguous shot boundaries (such as dynamic commercials and TV program lead-in & out clip [12]) in a long video stream. In order to robustly represent video segments, we introduce a set of spatial-temporal features as well as the color range features to construct a video signature. The obtained signatures tend to depict the video segment globally rather than focusing on its sequential details as key-frames based representation.

In consideration of the successful story of mrkd-tree in terms of searching high dimensional space [10] [11], we employ it to construct the index structure of the chopped segments. As mrkd-tree supports different query strategies such as *point query*, *range query* and *k-NN query* [5], it is suitable to various applications such as video copy detection, video content identification and retrieval. There have been some index based video similarity searching schemes, such as VQ index [7] and hash index [8]. However, their purpose is for *approximation K-NN* search where missed detection is probabilistically unavoidable. Our mrkd tree index emphasizes the *exact K-NN* query that is able to search out all the closest points of the given query, namely the K most similar video segments without any missed detection and false alarm.

2. OVERVIEW

Figure 1 illustrates our video similarity search scheme based on mrkd-tree structure. An index structure of the video database can be built off-line before querying process. Compared with traditional video similarity search method, our approach has the following features:

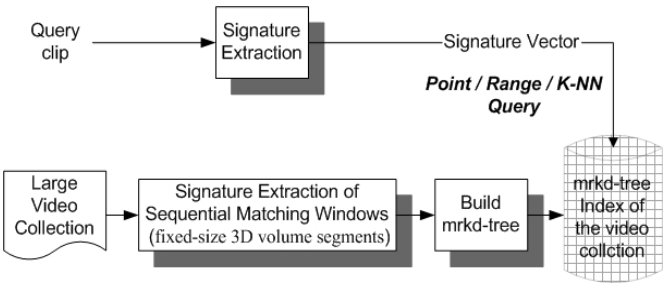

Figure 1. System Diagram

- In contrast to fast sequential search scheme applying **temporal pruning** [1], searching based on the index structure employs **spatial pruning** to accelerate the speed. The prior query results are always the most similar contents. A user can decide whether to do further searching according to the available results. Exhaustive seeking on the whole database is thus avoided.
- Instead of using representative images to characterize video content, we employ spatio-temporal feature and cumulative color histogram as a compact and robust signature to describe video content. Complex training phase (e.g. Vector Quantization employed in [1]) is unnecessary for extracting these signatures. Good performance has been proved in the experiment.
- As the index of video database is built for those temporally overlapped video segments with a predefined length, this scheme is also applicable to search for a sub-shot, a shot, and a series of shots. And by using different query strategies (point, K-NN, and range query), our method has provided a more general framework to fulfill different user search requirements.

3. Proposed Scheme

3.1 Problem Formulation

The problems of *exact K-NN query* and *range query* are formulated as follows. Let *DB* represent the video database containing temporally chopped video segments W_P, namely a series of matching windows, where $_P$ is the temporal position of a segment (matching window). Let *Q* denote the query clip with the same length as W_P. $D(\cdot,\cdot)$ and $S(\cdot,\cdot)$ are the function for dissimilarity and similarity measure between a query point and a matching window point in the feature space. Definition of $D(\cdot,\cdot)$ and $S(\cdot,\cdot)$ are given in Section 3.3. Now we have *exact K-NN query*:

$$KNNQuery(DB,Q,K) = \{w_{P1} \dots w_{PK} \in DB \,|\, \neg\exists w'_P \in DB \setminus \{w_{P1} \cdots w_{PK}\} \wedge \neg\exists i, 1 \le i \le K : D(w_{Pi},Q) > D(w'_P,Q)\} \quad (1)$$

Accordingly, range query is defined as:

$$RangeQuery(DB,Q,\varepsilon) = \{w_p \in DB \,|\, D(w_p,Q) \le \varepsilon\} \quad (2)$$

3.2 Signature Extraction of Video Segments

As one of the common visual features, color histogram is extensively used in video retrieval and identification [6] [9]. [9] applies compressed domain color features to form compact signature for fast video search. In [6], each individual frame is represented by four 178-bin color histograms on the HSV color space. And spatial information is incorporated by partitioning the image into four quadrants. Despite certain levels of success in [6] and [9], the drawback is also obvious, for example, color histogram is fragile to color distortion problems and it is inefficient to describe each individual key frame using a color histogram as in [6].

In consideration of the above features, Ferman et al. [4] presents various histogram-based color descriptors to reliably capture the color properties of video segments. Although such descriptors are reported to be robust and invulnerable to outlier frames within the shot, only color range information is considered, while both spatial information within each individual frame and temporal information are ignored. However our experiments show that spatial and color range information are both important for identification task.

Another type of feature which is robust to color distortion is ordinal feature proposed in [16]. Hampapur et al. [13] compared performance of using ordinal feature, motion feature and color feature respectively for video sequence matching. It was concluded that ordinal signature has the best performance. Based on this conclusion, we believe better performance could be achieved when combining ordinal feature and color range feature appropriately, with the former providing spatial information and the latter providing range information. Experiments in Section 4 will reveal these facts. As a matter of fact, many works such as [2] and [15] also incorporate the combined feature in order to improve the performance of retrieval and identification tasks.

Generally, the selection of the color feature and spatial-temporal feature as signature for identification task is motivated by the following reasons [19]:

(1) Compared with cost features such as edge, texture or refined color histogram, such as CCV (*color coherent vector*) [17] which also contain spatial information, they are inexpensive to acquire and could be estimated from MPEG compressed domain directly.

(2) Different from compact signature proposed in [8], these signatures still retain perceptual meaning. They are more robust to video formats variations.

(3) Ordinal feature is immune to global changes in the quality of the video and contain spatial information [13] [16], therefore it is a good complimentary to color feature.

3.2.1 Color Signature

We use the cumulative color histograms of all the sub-sampled I frames within a video segment as the color signature. For computational simplicity, the cumulative color distribution is estimated using the DC coefficients extracted from I frame in MPEG compressed video stream. The normalized cumulative histogram is:

$$H^{ccd} = \frac{1}{M}\sum_{i=b_k}^{b_{k+M-1}} H_i(j) \qquad j = 1,\cdots,B \tag{3}$$

where $H_i \big| i = b_k, b_{k+1}, \cdots, b_{k+M-1}$ denotes the color histogram of corresponding I frame within the video sequence. M is the number of I frames and B is the color bin number. In this paper, B is selected as 24, uniform quantization; for each channel c = Y, Cb and Cr, H_c^{ccd} is thus a *24-dimensional* feature vector. And the total size of the color signature H^{ccd} is *72-dimension*.

3.2.2 Spatial - Temporal Signature

The common drawbacks of color histogram descriptor lie in the lack of spatial information and its sensitivity to color variations [4]. In this section, we propose a spatial-temporal signature as compensation to the color signature. This signature contains both temporal and spatial information and is robust to color shifting, while remaining as compact as the color signature.

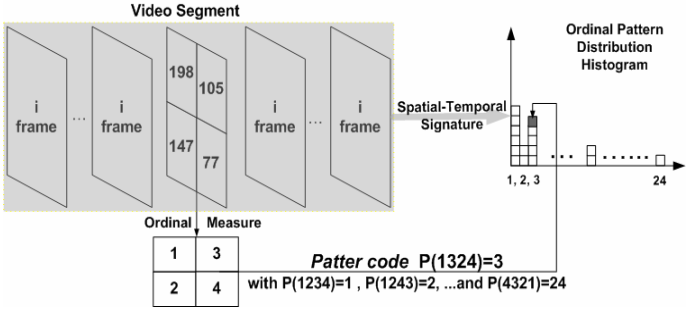

Figure 2. Spatial-Temporal Signature Description

As illustrated in Figure 2, each I frame is represented by a reduced image, of size 2×2. For each Y, Cb or Cr channel, we calculate the average value of each of the 4 sub-images by DC coefficients extracted from compressed domain directly. Raw feature extraction is then followed by the ordinal measure process. Each possible combination of ordinal measure result will be treated as an individual pattern. Therefore each I frame will be distributed a pattern code. All the patterns along the temporal axis are then accumulated to form a histogram. After the above operations, video segment can be compactly represented by 3 normalized 24-dimensional ***ordinal pattern distribution*** histograms, corresponding to Y, Cb, and Cr channels respectively.

For each channel c = Y, Cb, Cr, the video clip is represented as:

$$H_c^{opd} = (h_1, h_2, \cdots, h_l, \cdots, h_{NoP}) \quad 0 \le h_i \le 1 \quad and \quad \sum_i h_i = 1 \tag{4}$$

Here $NoP = 4! = 24$ is the dimension of the histogram, namely the number of possible patterns mentioned above. As a result, the total dimension of the spatial-temporal signature H^{opd} also becomes 72, the same size as the color signature.

3.3 Distance Metric

For both color signature and spatial-temporal signature, the distance of each Y, Cb and Cr channel is defined as Euclidean distance. The overall distance measure is defined as the linear combination of the **average** of spatial-temporal signature distance and the **minimum** of color signature distance among the Y, Cb, and Cr channels. And the overall similarity measure is defined as the reciprocal of distance value. Choosing such definition could further alleviate the affects of color shifting and other video variations. This is obtained by experimental comparison. In this paper, w is set to 0.5. We have the similarity definition between query and sliding matching widows as follows:

$$D_I^{opd}(H_Q^{opd}, H_{SW}^{opd}) = \frac{1}{3}\sum_{c=Y,Cb,Cr} D_c^{opd}(H_Q^{opd}, H_{SW}^{opd}) \tag{5}$$

$$D_I^{ccd}(H_Q^{ccd}, H_{SW}^{ccd}) = \min_{c=Y,Cb,Cr}\{D_c^{ccd}(H_Q^{opd}, H_{SW}^{opd})\} \tag{6}$$

$$D(H_Q, H_{SW}) = w \times D_I^{opd} + (1-w) \times D_I^{ccd} \tag{7}$$

$$S(H_Q, H_{SW}) = \frac{1}{w \times D_I^{opd} + (1-w) \times D_I^{ccd}} \tag{8}$$

3.4 Query Based on mrkd-Tree structure

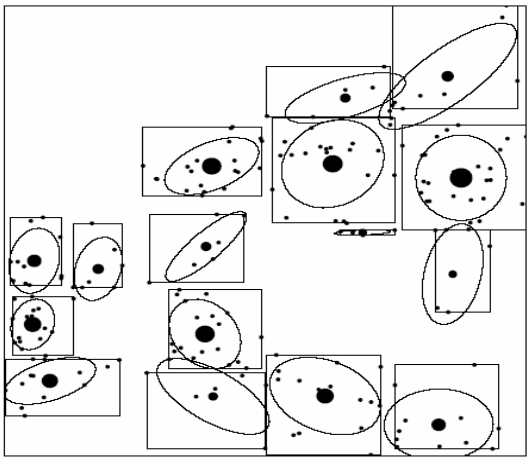

Figure 3. Example of an mrkd-Tree. The dots are the individual data points. The sizes and positions of the disks show the node counts and centroids. The ellipses and rectangles show the covariance and bounding boxes [20]

An mrkd-tree is introduced in [10] as an extension of kd-tree structure (Figure 3). The decoration is, at each node of kd-tree, it adds extra statistics about the node's data, such as their centroid, covariance and count. With such mrkd-tree structure, it is feasible to query the database with the same flexibility as a conventional linear search at greatly reduced computational cost. Since mrkd-tree structure adds sufficient statistics about all the data points below each node in the tree, it also has extended applications to speed up other operations such as locally weighted regression and mixture model based clustering and density estimation algorithms [11]. Here we use it as an effective index structure to complete *exact K-NN* query and *range* query.

By using an mrkd-tree structure for querying, at each node during the search, the algorithm considers whether it 1) can ignore all the points below that node because they are irrelevant to the current query, 2) can estimate the effects of all the points below that node without visiting them individually, or 3) must search further down the tree. The result is that most queries need only visit a small number of nodes in the tree to find out the nearest neighbors.

4. EXPERIMENTS

In our experiments, we apply both the fast sequential search and the index structure-based searching to complete fast video clip querying. An mrkd-tree was employed as the index structure, which can support different types of query such as *K-NN* query and *Range* query. These queries can be associated with most existing applications such as content-based retrieval and duplicate/near duplicate detection, etc. On the other hand, we implement the fast sequential scan method of "*active search*" proposed in [1] to compare it with an index structure-based query. Different from an index structure-based query, the active search requires seeking out the whole dataset to achieve the final results, no matter what query purpose it is. A performance comparison will be done in our experiments.

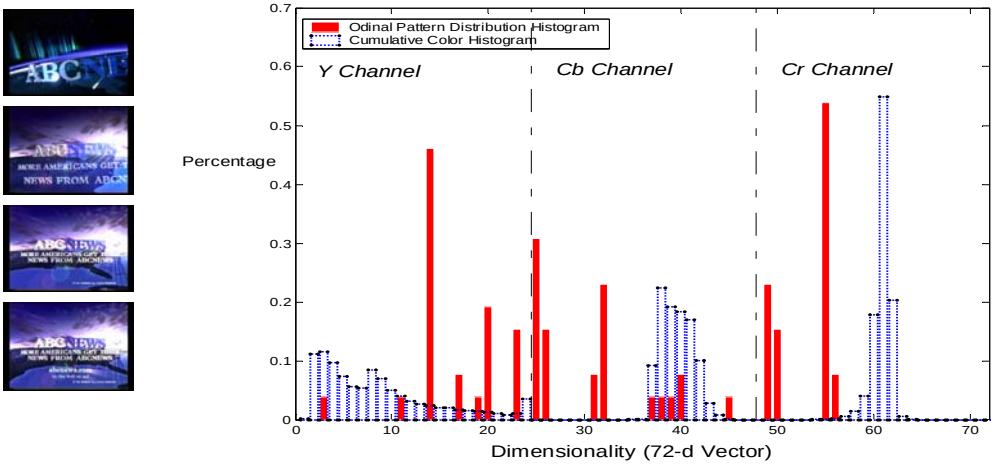

Figure 4. ABC News program lead-out clip (Left, 10 seconds) and its color and spatial-temporal signature representation (Right)

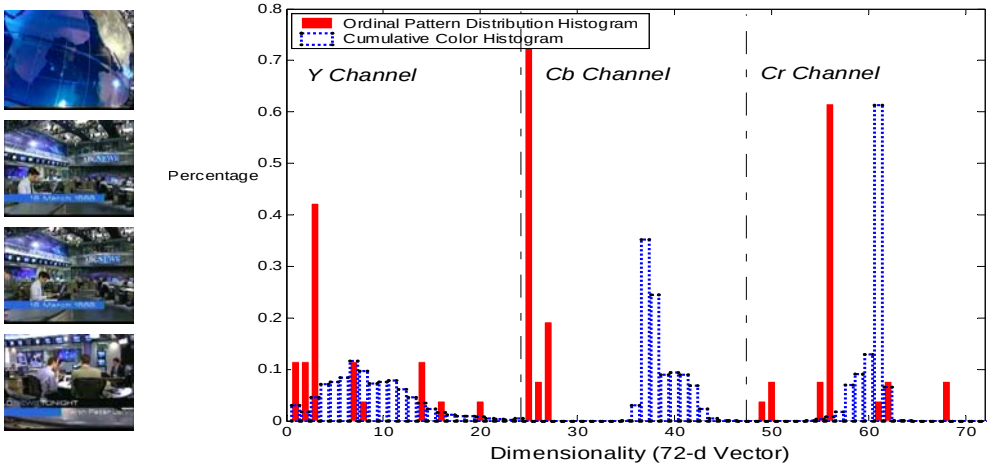

Figure 5. ABC News program lead-in clip (Left, 10 seconds) and its color and spatial-temporal signature representation (Right)

4.1 Dataset Description and Feature Extraction

The video database contains 22 streams of half-hour broadcast ABC news video obtained from TRECVID news dataset, encoded in MPEG1 at 1.5 Mb/sec and at frame rate of 29.97 fps. All of the video data are compressed with the GoP pattern IBBPBBPBBPBB, with I frame resolution of around 400ms. Among the 22 TV news videos, 12 of them are with image size of 352×240 and the other 10 with image size of 352×264. After feature extraction, all the half-hour segments are combined into one video sequence with the total length of 10.5 hours. An mrkd tree index will be built for this 10.5-hour sequence after it is chopped into segments of 10 seconds with an interval of around 0.4 second, the temporal distance of two neighbored I frames. Different from [3] which chops a long sequence into non-overlapped segments of 0.5 second, our chopped segments have an overlapped section of 9.6 seconds between the neighbored ones.

The query clip set contains a news lead-out clip in the ABC broadcast news video (Figure 4) and 78 commercials. The lead-out clip is at the length of 10 second and has 11 full instances in total and 2 incomplete instances in the 10.5-hours video dataset. The length of 78 commercials ranges from 10 to 60 seconds, and have in total 202 instances. Most of the instances have a slight color shifting compared with the original query. Our task is to fast and robustly identify and locate these instances inside the long video stream. We want to mention that the identification and retrieval of such repeated *non-news* sections inside a video stream helps to reveal the video structure. These sections include TV commercials, program lead-in/lead-out and other Video Structure Elements *(VSE)* which appear very often in many types of video to indicate starting or ending points of a particular video program, for instance, news programs or replay of sports video.

We first chop a video stream into a series of overlapping segments of fixed size. Both color signature and spatial-temporal signature are then extracted for each segment. Figure 4 and Figure 5 illustrate extracted signatures. The two examples are a news lead-out clip and a news lead-in clip in the ABC broadcast news video. Both of them are 10-sec long. As shown in Figure 4 and Figure 5, different from the accumulative color histogram representing the color density distribution, the proposed *ordinal pattern distribution* histogram (spatial-temporal signature) provides a unique sparse distribution, and thus it is more distinguishable than color range feature (color signature). For example, the lead-out (Figure 4) and the lead-in (Figure 5) clips have similar color signature because of similar dominant colors in the content; however, their spatial-temporal signatures are still quite distinguishable. In conclusion, as the ordinal pattern distribution is more unique and insensitive to a global color shifting or other color variations, this feature is a good supplementary to the color range feature towards a robust and composite feature set.

All the simulations were performed on a standard P4 @ 2.53G Hz PC (512 M memory). The algorithm was implemented in C++.

4.2 Query based on mrkd-tree Index

4.2.1 Exact K-NN Query Results

We only consider the news lead-out clip (10 seconds) for *K-NN query*. The distance metric is given in equation (7). The CPU cost of the pre-processing for querying is listed in Table 1. Note that for this 10.5 hour MPEG1 video dataset, only less than 0.5 hour is spent for feature extraction and building index.

In Table 2, CPU cost of *K-NN query* is given with different K. By using the NPT package [10], merely tens of milliseconds is needed to search out the very first best matching with the built index. While for the sequential searching, only when the whole database has been exhaustively searched out can it present the best matched result. To some extent, our proposed scheme provides a more flexible method for accessing the database and the user can determine whether further search is necessary according to available results. Take an example of video copy detection in large video set; current linear search method usually needs to exhaustively search out the whole database in order to claim no copy is detected. Nevertheless, with an index structure discussed above, we can apply *K-NN* search to fast locate the contents that are the most possible examples of the query according to the defined similarity. If the best matched result is not "similar" enough to be a copy, further searching is unnecessary. Computation is therefore greatly saved.

Table 1. CPU time cost for compressed domain feature extraction and mrkd tree index built (10.5 hours MPEG-1 video)

Process	CPU Time Cost (sec)	Size
Feature Extraction	1178.034	89,507 K Byte
mrkd Tree Built	470.610	18,301 Nodes of tree

Table 2. Exact *K-NN query* results using mrkd-tree

K-NN search / (feature space of 93,700 points)	K= 1	K= 10	K= 50	K= 100	K= 200	K= 1000	K= 5000
Time Consuming (sec)	0.078	0.204	0.969	1.563	2.641	5.078	20.365
Number of Contained Instances	1	2	7	11+1*	11+2*	11+2*	11+2*

** indicate the uncompleted instances*

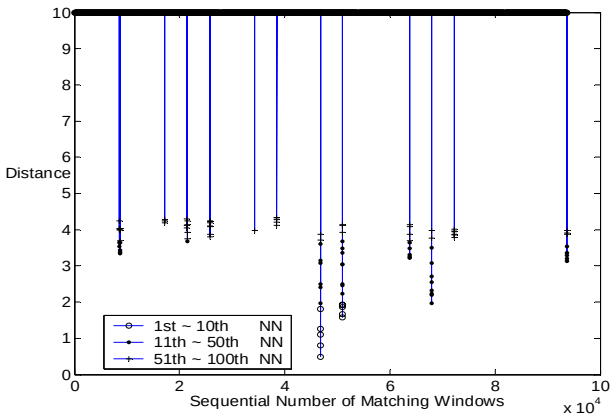

Figure 6. Searching results of *exact K-NN query* (K =100). Dataset consists of 10.5 hours MPEG-1 video sequence and the query length is 10-sec. The searching cost is 1.536 sec.

Figure 6 depicts the results of *K-NN query*. Since the original long video sequence is chopped into overlapped segments with the length of 10-seconds, each point in Figure 6 corresponds to a

matching window with the width of 10 seconds. Noted **not** each of the 100 retrieval points represents an individual instance. It is easy to understand that when one instance is found, its neighbors may also exhibit high similarity values due to the large overlap (9.6 sec/10 sec). Therefore only an area with clustered points can be claimed to be an instance from Figure 6. In other words, the number of detected instance should be counted by the peaks (12 peaks shown in Figure 6) instead of the number of retrieved points K.

4.2.2 Range Query Results

The 78 commercials are considered for Range Query to search out all the 202 instances in the 10.5-hour dataset. In order to be compatible with the built mrkd-tree, the first 10-sec section of each commercial is cut out as the query clip with the same length as the matching window. By utilizing mrkd-tree, the range searching cost of n points is *O(nlogn)* when ε is small. The performance of range searching is presented in Figure 7, by changing range ε adaptively.

In the experiment, we found that false alarms and missed detections are mainly caused by the I frame *shifted matching* problem, when the sub-sampled I frames of a given clip and that of the matching window are not well aligned at the temporal axis.

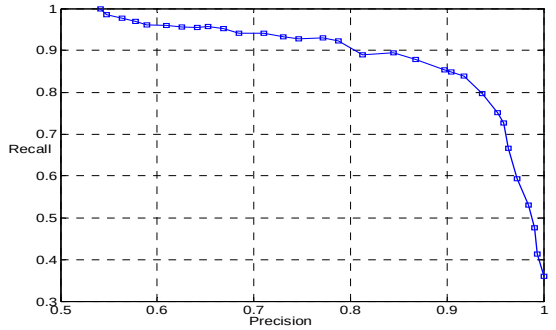

Figure 7. Searching results of Range query (*Precision = detects /(detects + false alarms)*) (*Recall = detects / (detects + miss detects)*)

4.3 Query based on Fast Sequential Search

In order to evaluate the performance of an index-structure based searching and justify our proposed new feature, we also implement the active search algorithm proposed in [1] for query acceleration. Different from the feature set proposed in [1], the complex training phase such as vector quantization is not required for our signature extraction. Nevertheless, our proposed signature is still compatible with the active search algorithm. The active search process is briefly introduced as follows:

Let the similarity metric array be $\{S_i; 1 \le i \le m+n-1\}$ corresponding to *m+n-1* sliding windows, where *n* and *m* are the total I frame number of given clip and target stream respectively. Based on [1] and [18], the search process can be accelerated by skipping unnecessary w_i steps.

$$w_i = \begin{cases} floor\ (\sqrt{2}\,N\,(\frac{1}{S_i} - \theta)) + 1 & if\ \ S_i < \frac{1}{\theta} \\ 1 & otherwise \end{cases} \qquad (9)$$

where S_i is the similarity value defined in equation (8); and its reciprocal is actually the distance value defined in equation (7). *N* is the number of I frames of the corresponding matching window. And θ is the predefined skip threshold.

After search, potential start position of the match will be determined by local maximum above the threshold, which fulfills the following conditions:

$$S_{k-1} \le S_k \ge S_{k+1} \quad and \quad S_k > \max\{T, m + k\sigma\} \qquad (10)$$

where *T* is the pre-defined preliminary threshold, *m* is the mean and σ is the deviation of the similarity curve; *k* is an empirically determined constant. Only when similarity value exceeds the maximum value of *T* and $m + k\sigma$, it can be treated as the detected instance. In our experiment, w in eq. (7) is set to 0.5, and θ in eq. (9) is set to 0.05. T in eq. (10) is 6.

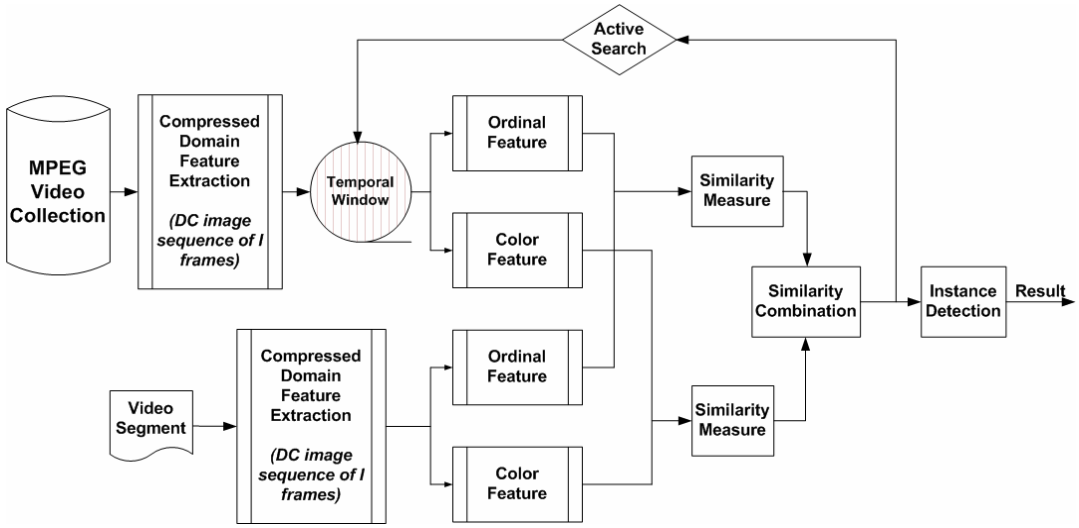

Figure 8. System chart of applying active search algorithm [1] with the proposed video signature

Figure 8 gives the system chart of applying our proposed signature to active search algorithm. Figure 9 presents the performance comparison when applying different features. It is clear that an appropriate combination of the color signature and the spatial-temporal signature can improve the performance against using them separately. Although we cannot achieve the accuracy of 100% with the proposed signatures, a comparable performance is still achieved with that in [12]. But their signature's size is 15 times larger than our proposed one.

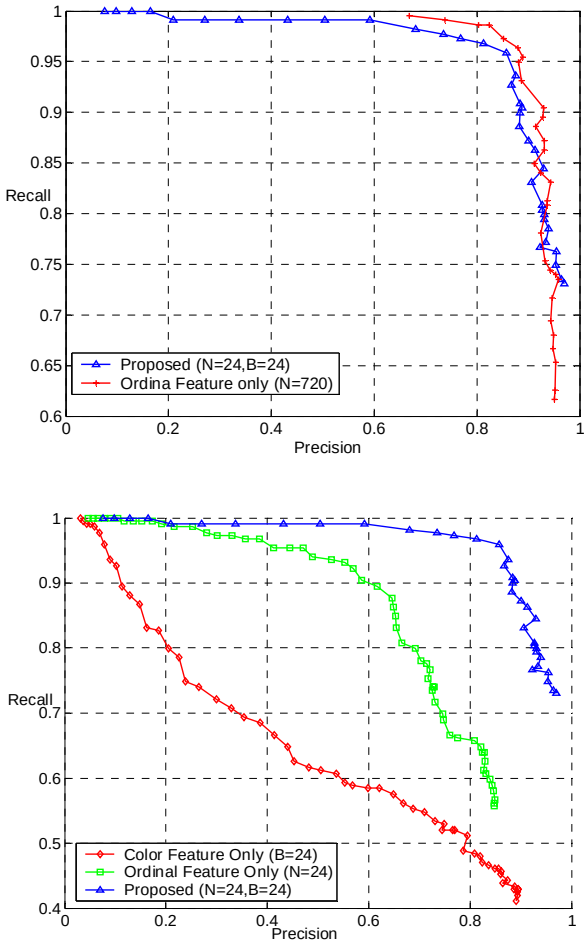

Figure 9. Performance Comparison using different features: proposed feature vs. 3× 720-d ordinal feature in [12] (Upper); proposed feature vs. 3× 24-d cumulative color feature and 3× 24-d ordinal feature respectively (Bottom); the detection curves are generated by varying the parameter k in eq. (10) (*Precision = detects /(detects + false alarms)*) (*Recall = detects / (detects + miss detects)*)

The main advantage of active search is the fast speed. In our experiments, in order to search through the whole 10.5 hour video, the active searching on average costs **0.011 second** when using the signatures extracted off-line [19].

5. CONCLUSION AND FUTURE WORK

In this paper, we deal with the video searching in large video collection as a sub-pattern matching problem and employ an index structure to accelerate querying process. A *multi resolution kd-tree* (*mrkd* tree) is used to construct an index for the temporally overlapped matching windows. Compared with the fast sequential scan searching, the mrkd-tree data structure provides an efficient mechanism for examining only those points closest to the query, thereby greatly reducing the computation towards the best matches. From our results, we found both an index based search (using *NN query*) and the active search are very fast to search out the very first instance of the query (i.e. searching out the 10.5 hour video in tens of milliseconds with the features extracted off-line). However, when more instances are needed, the active search is expected to be more efficient than the index based search in terms of CPU cost. Unlike the key-frames based shot matching, our method is based on elementary video segment of fixed-size, which could be a sub-shot, a shot or several shots. Since the research in high-dimensional index structure has been very active and fruitful over the past few years, many existing index structures (such as mrkd-tree, etc.) can support different types of queries. These various query strategies (such as *K-NN query*, *range query*) can reasonably produce different applications such as retrieval, duplicate/near-duplicate detection while avoiding going though the whole dataset. Therefore this can fulfill different user requirements in a more general framework. In this paper, we only give simple applications of using exact *K-NN query* and *range query* to identify video content in the sequence. How to select appropriate K or $\mathcal{E}$ for querying and how to use different query strategies to fulfill other applications such as video clip copy detection and content based retrieval will be our future work. Moreover, it might be expected to make the index structure accessible to queries with various lengths. More experiments will be done including the detection of dynamic commercial, flying logo detection etc. from more extensive video collections.

6. REFERENCES

[1] K. Kashino et al., "A Quick Search Method for Audio and Video Signals Based on Histogram Pruning," In *IEEE Trans. on Multimedia*, Vol. 5, No. 3, pp. 348-357, Sep. 2003

[2] A. K. Jain et al., " Query by video clip," In *Multimedia System*, Vol. 7, pp. 369-384, 1999

[3] D. DeMenthon et al., "Video retrieval using spatio-temporal descriptors," In *Proc. of ACM MM'03*, pp. 508-517, 2003

[4] A.M. Ferman, et al., "Robust color histogram descriptors for video segment retrieval and identification," In *IEEE Trans. on Image Processing*, vol. 1, Issue 5, May 2002

[5] Christian Bohm, et al., "Searching in high-dimensional spaces – index structures for improving the performance of multimedia databases," In *ACM Computing Survey (CSUR)*, vol.33, no.3, p.322-373, September 2001

[6] S.Cheung and A. Zakhor, "Efficient video similarity measurement with video signature," In *IEEE Trans. on Circuits and System for Video Technology*, vol. 13, pp. 59-74, 2003

[7] E. Tuncel et al., "VQ-index: an index structure for similarity searching in multimedia databases," In *Proc. of ACM MM'02*, pp. 543-552. Juan Les Pins, France, December 2002.

[8] J. Oostveen et al., "Feature extraction and a database strategy for video fingerprinting," In *Visual 2002, LNCS 2314*, pp. 117-128, 2002

[9] M.R. Naphade et al., “A Novel Scheme for Fast and Efficient Video Sequence Matching Using Compact Signatures,” In *Proc. SPIE, Storage and Retrieval for Media Databases 2000*, Vol. 3972, pp. 564-572, 2000

[10] K. Deng and A.W. Moore, “Multi-resolution instanced based learning,” In *Proc. of Int. Joint Conf. on Artificial Intelligence’95*, pp. 1233-1239, San Francisco, Morgan Kaufmann, 1995. NPT package is available at http://www.autonlab.org/astro/npt/index.html

[11] A.W. Moore, “Very fast mixture-model-based clustering using multi-resolution kd-trees, In *Advances in Neural Information Processing System 10*, pp. 543-549, April 1999

[12] Junsong Yuan, Qi Tian and S. Ranganath, “Fast and Robust Search Method for Short Video Clips from Large Video Collection,” in *Proc. of ICPR’04*, Cambridge, UK, August 2004

[13] A. Hampapur, K. Hyun, and R. Bolle., “Comparison of Sequence Matching Techniques for Video Copy Detection,” In *SPIE. Storage and Retrieval for Media Databases 2002*, vol. 4676, pp. 194-201, San Jose, CA, USA, Jan. 2002.

[14] L. Chen and T.S.Chua, “A match and tiling approach to content-based video retrieval,” In *Proc. of ICME’01*, pp. 301-304, 2001

[15] V. Kulesh et al., “Video clip recognition using joint audio-visual processing model,” In *Proc. of ICPR’02*, vol. 1, pp. 500-503, 2002

[16] D.N. Bhat, S.K.Nayar, “Ordinal measures for image correspondence,” In *IEEE Trans. on PAMI*, Vol. 20, No. 4, pp. 415-423, 1998

[17] G. Pass et al., “Comparing images using color coherence vectors,” In *Proc. of ACM Multimedia’96*, pp. 65-73, 1996

[18] Akisato Kimura, et al., “A Quick Search Method for Multimedia Signals Using Feature Compression Based on Piecewise Linear Maps,” In *Proc. of ICASSP’02*, Vol. 4 , pp. 3656 -3659, May 2002

[19] Junsong Yuan, Ling-Yu Duan, Qi Tian, “Fast Video Segment Identification from Large Video Collection,” To appear in 2004 Pacific-Rim Conference on Multimedia (PCM’04), Nov. 2004, Japan

[20] Alexander G.Gray and Andrew W.Moore, “Nonparametric Density Estimation: Toward Computational Tractability,” In *SIAM International Conference on Data Mining’03*, San Francisco, CA, USA