Istanbul Technical University at TRECVID 2008

O. Gursoy, B. Gunsel

Multimedia Signal Processing and Pattern Recognition Group
Department of Electronics and Communications Engineering
Istanbul Technical University
Maslak/Istanbul, 34496, TURKEY
ozan_gursoy0@yahoo.com, gunselb@itu.edu.tr

ABSTRACT

ITU MSPR Group participates the TREC Video Retrieval Evaluation (TRECVID) in Content Based Copy Detection (CBCD) task. The system proposed by ITU MSPR consists of two main modules: Extraction of video fingerprints and search/retrieval. We propose a feature extraction scheme based on the Nonnegative Matrix Factorization(NMF)[1], which is an efficient dimension reduction technique in video processing[2]. Video fingerprint generation module takes the factorization matrices generated by NMF as its input and converts them to binary hashes by differencial coding. Extracted hashes are indexed into a database. Searching module first applies a hash matching procedure to locate potential matching points. It is followed by temporal merging that eliminates false alarms while combining subsegments. Initial results are promising for insertion of pattern, reencoding, blurring, change of gamma and noise addition. Future work will include impoving the current results and searching for robustness to geometric transformations such as shift, crop, flip and picture-in-picture.

I. INTRODUCTION

As a first time participant of TRECVID, we propose a Video Fingerprinting system in the context of TRECVID 2008 Content Based Copy Detection task. Conventionally a copy detection task forces a video fingerprinting system to be robust to transformations without altering the content but preserving the uniqueness of it. These transformations include insertion of pattern, reencoding, blurring, change of gamma, addition of noise and resizing.

Fig. 1 shows a general block diagram of the system proposed by our group. First module extracts features by using Nonnegative Matrix Factorization (NMF). In [3] we have shown that the NMF, with its ability to reduce dimension and extract intuitive features in an efficient and simple way, is a powerful content representation technique. This is mainly because of the additivity property of NMF that provides bases to capture local components of the content. Our previous work on video content representation by incremental subspace learning [3, 4] have driven us to benefit from NMF in copy detection task.

After feature extraction, the NMF based features are converted to binary hashes and form video fingerprints. Second module of the proposed system indexes and stores the hash values to form a video fingerprinting database. Searching module takes extracted fingerprints of a query video as inputs to match with the fingerprints of reference video clips in a video fingerprinting database. Matched results are ranked by a similarity measure. Video name and matching locations in both query and reference videos of retrieved results are reported by a GUI.

This paper is organized as follows. Section II gives the mathematical background and summarizes the proposed feature extraction and video fingerprinting scheme. Our search and retrieval system is described in Section III and details of the graphical user interface is given. Section IV addresses the experimental results and conclusions.

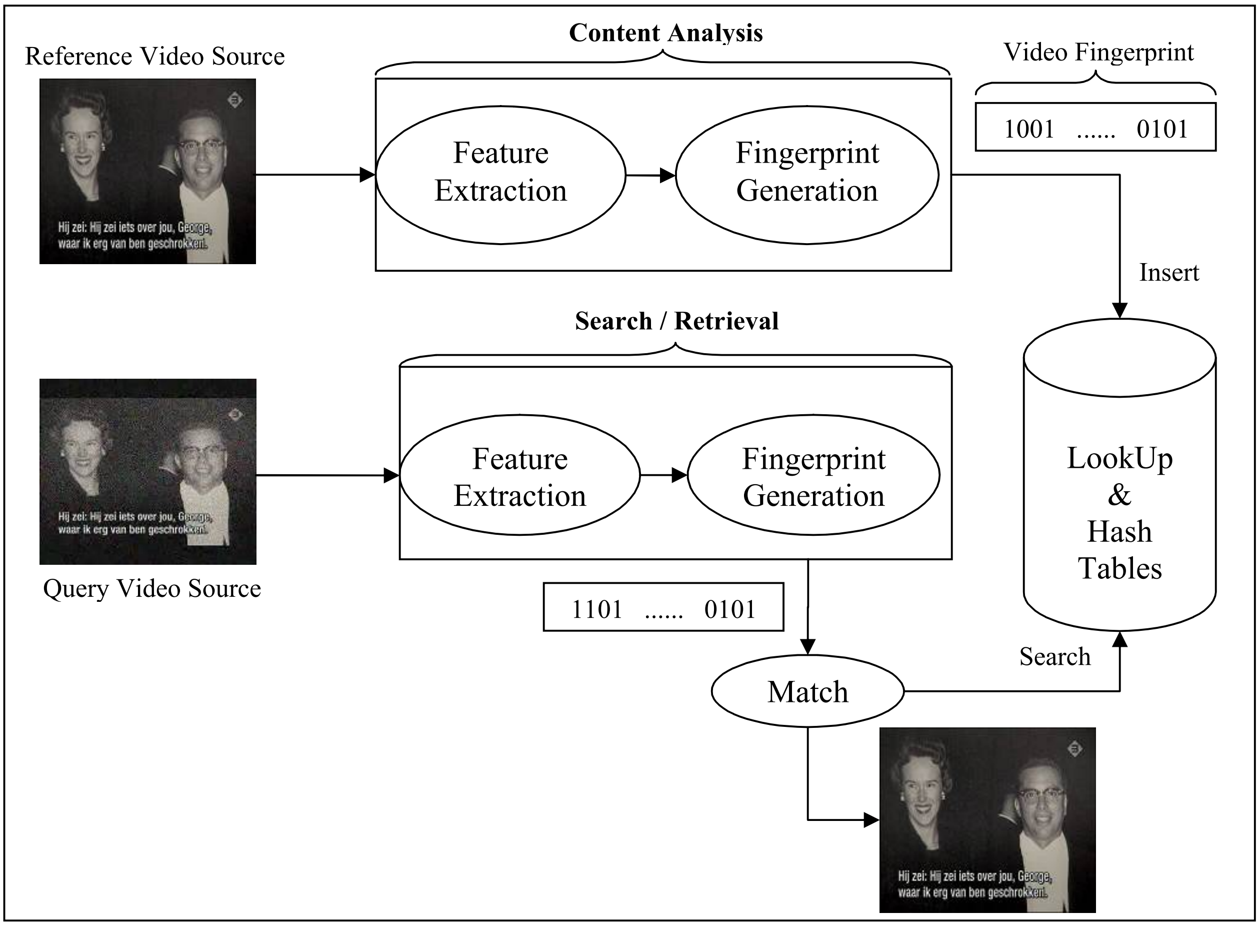


Fig. 1: System Overview

II. EXTRACTION OF VIDEO FINGERPRINTS

Our feature extraction approach is based on individual frames. Each frame in a video clip is factorized by Nonnegative Matrix Factorization (NMF) to its characteristic matrices, **W** and **H**. The computed **W** and **H** matrices for each frame are used as the features extracted from a video clip.

As formulated in Eq. (1), NMF represents a data matrix as a multiplication of two matrices, which are computed by a gradient descent based updating rule that minimizes the reconstruction error given by Eq. (2). The update rules for **W** and **H** are given by Eq. (3) where *t* refers to the iteration number, *T* denotes the transpose, *a* = 1,2,…,*r* ; *i* = 1,2,….,*n* and *j* = 1,2,…,*m*.

$$\mathbf{V} \approx \mathbf{WH} \tag{1}$$

$$F = \| \mathbf{V} - \mathbf{WH} \|^2 = \sum_{i=1}^{n} \sum_{j=1}^{m} \left(V_{ij} - (\mathbf{WH})_{ij} \right)^2 , \tag{2}$$

$$H_{aj}^{t+1} = H_{aj}^{t} \frac{\left(\mathbf{W}^{t^T} \mathbf{V} \right)_{aj}}{\left(\mathbf{W}^{t^T} \mathbf{W}^{t} \mathbf{H}^{t} \right)_{aj}} , \qquad W_{ia}^{t+1} = W_{ia}^{t} \frac{\left(\mathbf{V} \mathbf{H}^{t+1^T} \right)_{ia}}{\left(\mathbf{W}^{t} \mathbf{H}^{t+1} \mathbf{H}^{t+1^T} \right)_{ia}} . \tag{3}$$

The data matrix $\mathbf{V} \in \mathrm{R}^{nxm}$ is formed by banding *m* observation vectors together as the columns of **V**. $\mathbf{W} \in \mathrm{R}^{nxr}$ matrix contains the *r* constructing bases vectors and the $\mathbf{H} \in \mathrm{R}^{rxm}$ matrix consists of the weighening coefficients associated with the bases vectors in **W**. In our approach, the data matrix **V** is taken as a single video frame whose columns are both, in a manner, similar to each other and also form a characteristic pattern. We can assume that neighbouring pixels are similar but also the content in a frame has a variation as the pixels are scanned through the frame.

In NMF factorization, the dimension reduction is achieved by the rank parameter *r*. As the purpose of our task is detecting the copy contents, which may have severe transformations, we select a low rank value, 2. This leads our features to be more robust to transformations without losing the distinctive content.

To decrease the computational complexity, we resize the video frames. As shown in our previous works, using the 8 by 8 block means instead of all the pixels of a frame, is not affecting the representative power of our features. As a result of this we have 36 by 2 **W** matix and 2 by 44 **H** matrix, total of 160 feature points from 323x288 = 101376 pixels.

After extraction of feature matrices **W** and **H**, we convert them to binary hashes to construct our video fingerprints. Constructed video fingerprints must be robust to transformations and must not be fragile as cryptographic hashes. Similar to [5], we differentially code the transformation matrices' elements of consecutive two frames.

Eq. (4) shows the calculation of G_W^k and G_H^k bitarrays using the W^k, H^k and W^{k+1}, H^{k+1} matrices. Lower indices show the matrix elements and the upper indices show the frame numbers. Indice values are changed as $a = 0,1,\dots,m-1$; $b = 0,1,\dots,n-1$ and $j = 0,1,\dots,r-2$. *sgn[x]* function returns 0 or 1 according to the sign of it's parameter *x*. Spatial difference is calculated using corresponding elements of neighbouring columns of W^k matrix. Same procedure is applied to the rows of H^k matrix. The final hash value of frame *k*, G^k, can be constructed from G_W^k or G_H^k or from the combination of both G_W^k and G_H^k. Bit length of G^k determines fragility and robustness of the hash.

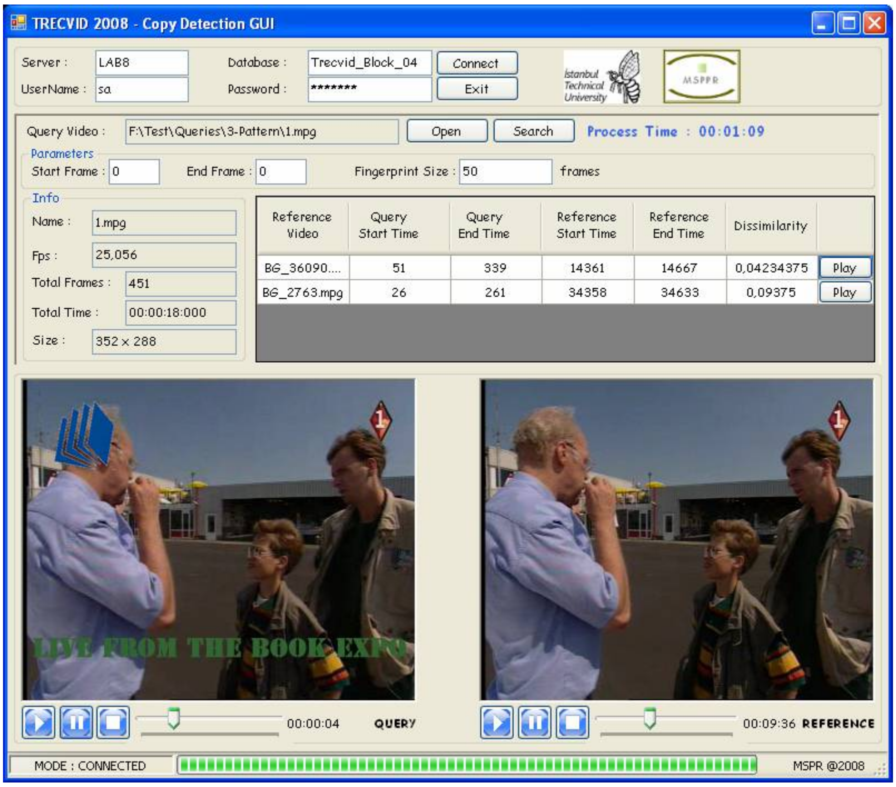
$$\begin{aligned} G_W^k\left[a+m\cdot j\right] &= \operatorname{sgn}\left[\left(W_{a,j}^k - W_{a,j+1}^k\right) - \alpha\left(W_{a,j}^{k+1} - W_{a,j+1}^{k+1}\right)\right] \\ G_H^k\left[b+n\cdot j\right] &= \operatorname{sgn}\left[\left(H_{j,b}^k - H_{j+1,b}^k\right) - \alpha\left(H_{j,b}^{k+1} - H_{j+1,b}^{k+1}\right)\right] \end{aligned} \qquad (4)$$

When consecutive frames are belong to a still scene, in which frames do not change remarkably, temporal difference is completely determined by noise. In Eq. (4), $\alpha=0.95$ value is used to remove such an unreliable effect on hash values.

III. SEARCH AND RETRIEVAL USER INTERFACE

Hash values of a reference video clip are extracted and stored in a database by an offline process. Instead of searching all possible locations, search is performed on potention locations which are extracted from a lookup table using the hash values of the query video. The lookup table holds every occurance of each hash value in the database by a pair of values: "Video Id" and the index of the hash value in that video. It is expected that at least one hash value of the query video clip remains unchanged after the transformations to perform a comparison, otherwise no search locations can be found. Matching is performed over a window, **s**, of hash values. Hash matching procedure is followed by temporal merging that eliminates false alarms while combining subsegments. Matching results are ranked by their similarity to the query video clip. Similarity is measured based on the normalized Hamming distance between matched hash values.

A graphical user interface for search and retrieval can be seen in Fig. 2. The development environment for the GUI is Visual Studio 2005 C++/C#. MsSql Server 2000 Developer Edition is used for databases. OpenCv Library is used for matrix operations. User interface enables to connect to different pre-built video fingerprinting databases. After a database connection is established, user starts his/her search by selecting a query video. Search can be made on a user defined segment of the query video if user is not interested in the whole query video. Search window size is another option that can be adjusted by the user. When searching ends, a list of retrieved results are displayed in a grid. The reported results include the reference video name, start and end times of matching segment in the query video together with the corresponding start and end times in the reference video. The results are sorted by a similarity measure which is calculated during the searching process. A play button for each matching result enables user to play reference and query video segments at the same time, beginning from the reported start times. This option makes it easier to watch and decide whether it is a correct match or not. Processing time for the whole search and retrieval process is displayed to the user.



TRECVID 2008 - Copy Detection GUI

Server : LAB8
UserName : sa
Database : Trecvid_Block_04
Password : *******

Query Video : F:\Test\Queries\3-Pattern\1.mpg

Process Time : 00:01:09

Parameters
Start Frame : 0 End Frame : 0 Fingerprint Size : 50 frames

Info
Name : 1.mpg
Fps : 25,056
Total Frames : 451
Total Time : 00:00:18:000
Size : 352 x 288

Reference Video	Query Start Time	Query End Time	Reference Start Time	Reference End Time	Dissimilarity	
BG_36090....	51	339	14361	14667	0,04234375	Play
BG_2763.mpg	26	261	34358	34633	0,09375	Play

00:00:04 QUERY

00:09:36 REFERENCE

MODE : CONNECTED

MSPR @2008

Fig. 2: Graphical User Interface for search and retrieval

IV. EXPERIMENTAL RESULTS AND CONCLUSIONS

In this section, we will present our resultswhich are based on multiple decision rules. Each frame is coded by 40-44 bits and the search window varies from 4 to 6 seconds. Experimental results for each transformation are reported seperately on the columns of Table 1. Our algorithm has promising results for insertion of pattern, reencoding, change of gamma, blurring, noise addition, resizing, frame drop and change of contrast type of transformations. The results of picture-in-picture and geometric transformations at Attack 8 thorough 10 need to be improved.

	Camc.	PicInPic	Pattern	Reenc.	Gamma	Att6	Att7	Att8	Att9	Att10
Match	6	0	39	74	95	71	44	6	0	2
Miss	128	134	95	60	39	63	90	128	134	132
False	88	150	119	92	88	135	147	83	75	66

Table 1: Multiple decision rule based results for each transformation

The values in Table 1 will be evaluated by Normalized Detection Cost Rate (NDCR) which is defined in Eq. (5). The reason to define NDCR instead of conventional precison and recall is to avoid dependency on class distributions of the test data. One third of the test query videos do not contain a reference video.

$$NDCR = P_{Miss} + \frac{C_{FA}}{C_{Miss} \cdot R_{Target}} \cdot R_{FA} \qquad (5)$$

In Eq. (5), P_{Miss} and R_{FA} are the conditional probability of a missed copy and the false alarm rate respectively; C_{Miss} and C_{FA} are the costs of a Miss and a False Alarm, respectively; and R_{target} is the a priori target rate. The parameters are defined by TRECVID as $R_{target} = 0.5$/hr , $C_{Miss} = 10$ and $C_{FA} = 1$.

In our first participation in the TRECVID evalualtions, our goal was to develop a robust, content based video fingerprinting system that allows fast and efficient search. Initial results are promising especially on non-geometric transformations.

We are currently extending the range of our experiments and will present our results during the meeting. Basically we are running the same scheme on 9 equal size blocks rather than the whole frame. We expect to achieve robustness to some of the transformations by working on individual blocks. These include picture-in-picture, insertion of pattern, cropping and shifting type of transformations. Because block-based indexing gives us the flexibility in searching in such a way that randomly accessing to different blocks having different confidence levels. We basically assign high confidence levels to different blocks that adaptively adjusted for each transformation. For example we can simply select the central block, which most of the time contains more knowledge compared to border blocks, or we can find the block in which content changes higher than the rest. Currently we are using Hamming distance to measure the dissimilarity between different blocks.

V. REFERENCES

[1] D.D. Lee and H.S. Seung, Learning the parts of objects by nonnegative matrix factorization, Nature, 401 (1999) 788-791.

[2] S.S. Bucak, B. Gunsel, Incremental subspace learning via non-negative matrix factorization, Pattern Recognition (accepted)

[3] S.S. Bucak, B. Gunsel, Video content representation by incremental non-negative matrix factorization, in: IEEE Int. Conference on Image Processing, San Antonio, USA, 2007, pp. 113-116.

[4] S.S. Bucak, B. Gunsel, O. Gursoy, Incremental non-negative matrix factorization for dynamic background modeling, in: ICEIS Int. Workshop on Pattern Recognition in Information Systems, Funchal, Portugal, 2007, pp.107-116.

[5] J. Oostveen, T. Kalker, J. Haitsma, Feature extraction and a database strategy for video fingerprinting, in: Int. Conf. on Recent Advances in Visual Info. Systems, Taiwan, 2002, pp. 117-128.