ITU MSPR TRECVID 2010 VIDEO COPY DETECTION SYSTEM

Sezer Kutluk, Bilge Gunsel

Multimedia Signal Processing and Pattern Recognition Group
Department of Electronics and Communications Engineering
Istanbul Technical University
Maslak/Istanbul, 34496, Turkey
kutluks@itu.edu.tr**,** gunselb@itu.edu.tr

Abstract

In this paper we describe the system designed by the ITU MSPR Group for content based video fingerprinting as applied to the TRECVID 2010 Content Based Copy Detection (CBCD) benchmark. This year focus of the system was on integration of audio and video fingerprinting to improve the robustness to attacks. The proposed system consists of three main modules: Audio/video fingerprint extraction, audio/video search and retrieval, and audiovisual decision fusion. We propose a video feature extraction scheme based on the Nonnegative Matrix Factorization (NMF) which is an efficient dimension reduction technique in video processing. Video fingerprint generation module takes the factorization matrices generated by NMF as its input and converts them to binary hashes by differencial coding [1, 2]. For audio data we perform an audio fingerprinting method that is similar to the one proposed in [3]. Extracted audio and video hashes are indexed into a database. Searching module first applies a hash matching procedure to locate potential matching points both in audio and video. This is followed by decision fusion that eliminates false alarms and finalizes the matching and retrieval.

1. Introduction

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 geometric and global attacks for video including simulated camcording, insertion of pattern, reencoding, blurring, change of gamma, addition of noise, resizing, cropping, shifting, change of contrast, text insertion, flipping and frame dropping. In terms of audio attacks, MP3 compression, multiband companding, bandwith limiting, single-band companding, mixing with speech, multiband compressing and bandpass filtering need to be considered.

This year the ITU MSPR Group has participated in the TRECVID Content Based Copy Detection Track, and submitted results for the audiovisual video fingerprinting. This paper presents the ITU MSPR system and reports the test results. It is shown that the designed system is highly robust to global attacks while it needs to be improved against the severe geometric attacks. It is also shown that audiovisual decision making module signaficantly improves the system performance especially for the picture-in-picture type queries.

2. Designed Copy Detection System

Figure 1 illustrates block diagram of the designed video fingerprinting system that consists of two main modules, i.e., the indexing module, search/retrieval module and their interaction with the fingerprint and video databases. A similar block scheme can be given for the audio fingerprinting. The designed copy detection system integrates outputs of the video and audio fingerprinting modules to provide temporarily matched video clips that are retrieved by audiovisual queries.

2.1. Fingerprint Extraction and Indexing

As it is shown in Fig.1, we divide a video clip into time intervals of length T_i sec. Frames in an interval are resized and divided into blocks. Using these resized and divided frame parts, a hash value related to that time interval is generated. These hash values are then binarized, indexed and inserted into the video fingerprint database.

Extraction of fingerprints from a reference video clip and storing them in a database is an offline process. Since these hash values are extracted using all the frames in an interval T_i, time information is still kept in the database. Similarly, extracted hash values from all blocks are stored in the database, so block relations are still kept. This structure allows us to use any combination of these hash values for fingerprint generation. Different paths can be chosen to generate fingerprints upon the query video.

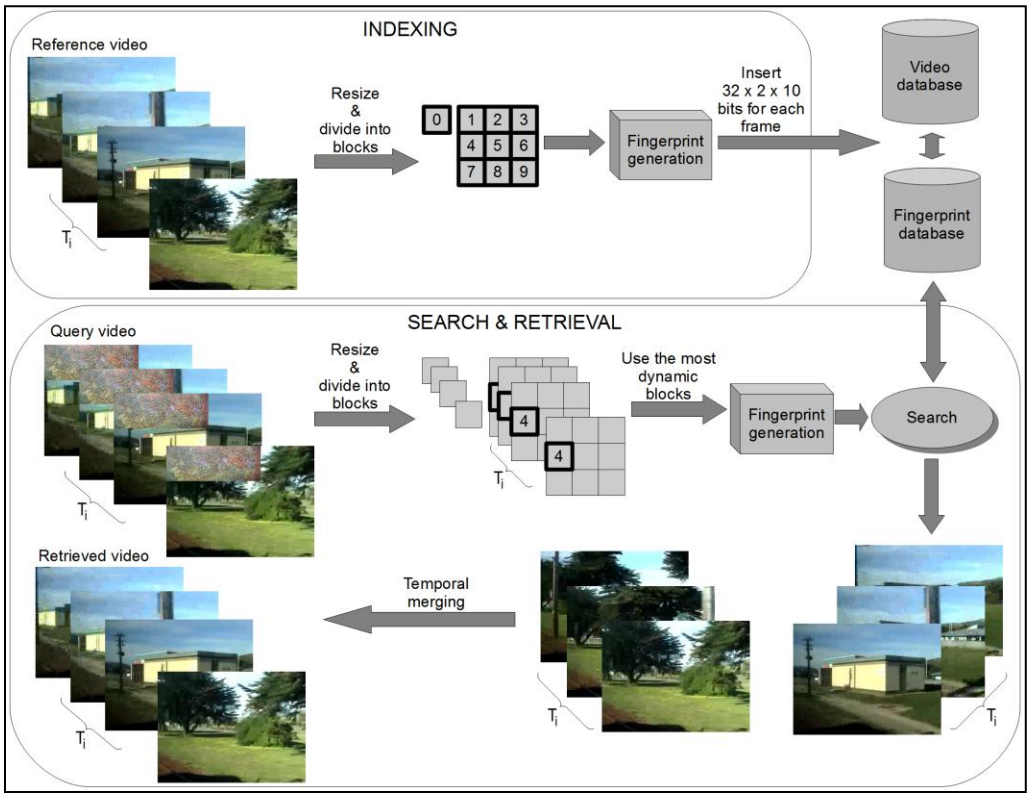


Fig. 1: Video fngerprinting system oerview.

Since the block-based processing of video may provide robustness to attacks like insertion of pattern, picture in picture, cropping and shifting, we developed a block-based indexing system. Furthermore, it is expected to minimize the computational complexity with such a block-based hashing scheme. Another advantage of using a block based indexing scheme is to minimize the collusion probability arising from extremely stable scenes. Figure 2 displays the implemented block-based fingerprint extraction scheme in detail. A video frame is first scaled into two different sizes, which are 32 x 32 pixels and 96 x 96 pixels. The idea behind resizing is it is common to use DC images that reflects the original content with a much more small number of pixels that yields a significant decrease in computational complexity. The 96 x 96 pixels-sized frame is then divided into 9 non-overlapping blocks. The resized frame and 9 blocks are factorized, and NMF representations, W^{kb} and H^{kb}, are achieved. Here, k is the frame index, and $b = 0,\ldots,9$ is the block index. When $b = 0$, the 32 x 32 pixels-sized frame itself is used as a block.

Totally, there are 10 blocks. Figure 2 shows how W^{kb} and H^{kb} matrices are achieved. Binary subfingerprints are achieved by differential coding of NMF representations of the same blocks of two consecutive frames. For each frame, 32 x 2 x 10 = 640 bits are inserted into fingerprint database.

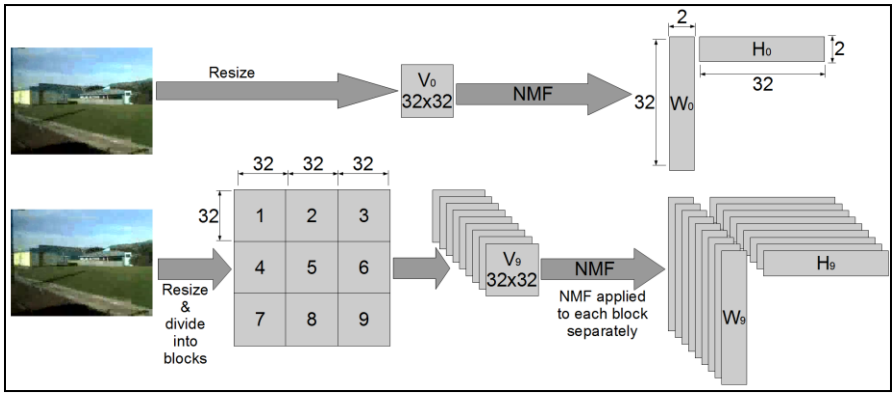


Fig. 2: Extraction of video hashes.

2.1.1. Mathematical Formulation

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 matrix transpose, $a = 1,2,\dots,r;\ \ i = 1,2,\dots,n$ and $j = 1,2,\dots,m$.

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

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

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

The data matrix $V \in R^{nxm}$ is formed by concatenating m video frames together as the columns of V. $W \in R^{nxr}$ matrix contains the r constructing bases vectors and the $H \in R^{rxm}$ matrix consists of the weighting coefficients associated with the basis vectors in W. In our approach, the data matrix V is taken as a single video frame or a video block 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, the dimension reduction is achieved by the rank parameter r [4]. As the purpose of our task is detecting the original or copied 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.

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^{kb} and G_H^{kb} bit arrays using the W^{kb}, H^{kb} and $W^{(k+1)b}$, $H^{(k+1)b}$ 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 its parameter *x*. Spatial difference is calculated using corresponding elements of neighbouring columns of W^{kb} matrix. Same procedure is applied to the rows of H^{kb} matrix. The final hash value of frame k, G^{kb}, can be constructed from G_W^{kb} or G_H^{kb} or from the combination of both G_W^{kb} and G_H^{kb}. Bit length of G^{kb} determines fragility and robustness of the hash.

$$G_W^{kb}[i] = sgn\left[\left(W_{i,a}^{kb} - W_{i,a+1}^{kb}\right) - \alpha\left(\mathrm{W}_{\mathrm{i,a}}^{(\mathrm{k+1})\mathrm{b}} - \mathrm{W}_{\mathrm{i,a+1}}^{(\mathrm{k+1})\mathrm{b}}\right)\right]$$
$$G_H^{kb}[j] = sgn\left[\left(H_{a,j}^{kb} - H_{a+1,j}^{kb}\right) - \alpha(\mathrm{H}_{\mathrm{a,j}}^{(\mathrm{k+1})\mathrm{b}} - \mathrm{H}_{\mathrm{a+1,j}}^{(\mathrm{k+1})\mathrm{b}})\right] \quad (4)$$

When consecutive frames belong to a still scene, in which frames do not change remarkably, temporal difference is completely determined by noise. In Eq. (4), $\alpha = 0.9$ value is used to remove such an unreliable effect on hash values.

2.2. Search and Retrieval

Search and retrieval is performed as an online process. As it is shown in Fig.1, the query video has also been searched for the video units of length T_i sec. Similar to the indexing phase, the video frames in an interval are resized and divided into blocks. Using these resized and divided frame parts, a hash value related to that time interval is generated. These hash values are then binarized.

Instead of searching all possible locations, search is performed on potential locations which are extracted from a lookup table using the hash values of the query video. The look-up 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 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 the matched hash values.

Between two consecutive frames of a query video, two most dynamic blocks are selected from subfingerprints by using Hamming distance. These two subfingerprints are searched in the database. Beginning from the matched frame, subfingerprints from the same block for a part of video clip, which is of length T_i, are concatenated to constitute a fingerprint. Fingerprints generated from the reference and query video clips are matched whether the Hamming distance of them remains less than a threshold. Considering the various frame rates (fps), length of the video finperpringt is specified as (fps x 32 x 2) bits.

When the length of a query is longer than T_i, all overlapping parts of length T_i are used for search, and a list of all matches is recorded. Then a temporal merging process is applied to eliminate the false alarms and to decide whether a match is acceptable or not. The temporal merging is the last process of search and retrieval task, and it produces a list of retrieved reference videos that match the input queries, with

information of matching time intervals. A similar procedure is applied on the audio data. In audio query matching our granularity was 2 seconds to guarantee a consistent matching under attacks.

2.3. User Interface

Figure 3 illustrates the graphical user interface of the video search and retrieval module of ITU MSPR Copy Detection software. 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. FFmpeg library is used for decoding video files. User interface enables to connect to different pre-built video fingerprinting databases. After a database connection is established, user starts the 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 or written into a file. 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 video query matching results are sorted by a similarity measure which is calculated during the searching process. Processing time for the whole search and retrieval process is displayed to the user.

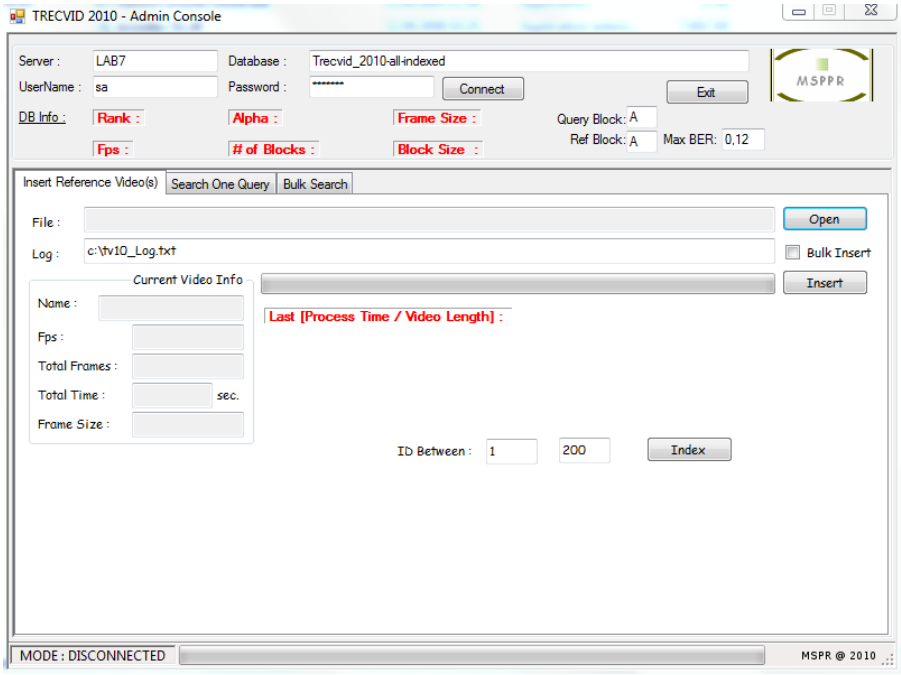


Fig. 3: GUI of the video search and retrieval module of ITU MSPR Copy Detection software.

Figure 4 illustrates the graphical user interface of the audio search and retrieval module of ITU MSPR Copy Detection software. This module can do search and retrieval over a pre-built fingerprint database. It allows insertion of new audio fingerprints into the database. After connection to database is established, individual or a list of queries are given as inputs for search and retrieval. The system allows search and matching at different granularities as well as at various acceptable matching thresholds. When the search operation is completed, the retrieved audio IDs along with the record name are displayed on the user

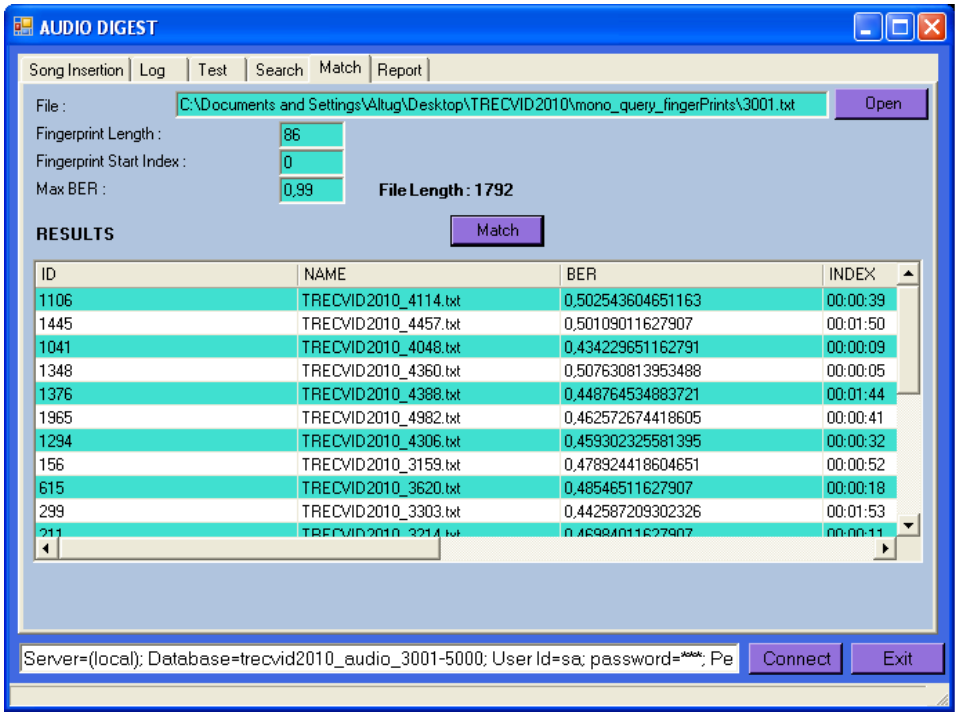


Fig. 4: GUI of the audio search and retrieval module of ITU MSPR Copy Detection software

interface as a grid, or optionally for batch processing, results are written into a file. Reference audio file name, query name, matching bit error rate score and matched time stamps (indexes) are reported.

In the developed system, fingerprinting and search/retrieval tasks are performed separately for audio and video data. Then a decision fusion procedure combines the retrieval results of audio and video data. While the audiovisual decision has been made, a number of cases should be considered. If the audiovisual query matching has been established for both video and audio, then a positive matching is declared. If the matching is established for either audio or video, then the decleration has been made based on the length of the match. Since the audio copy detection module has a lower false alarm rate than the video copy detection module, we give a higher priority to audio matching when the matching results provided by the audio and video modules mismatch.

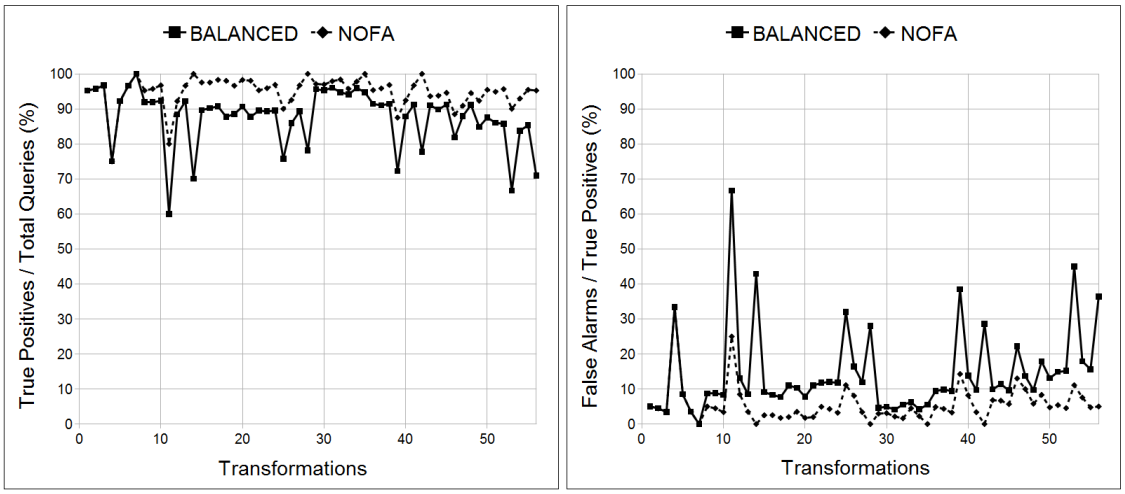
3. Test Results

We have focused on nearly 430 hours of reference videos, and 10976 audio-visual queries under Trecvid 2010 Content Based Copy Detection benchmarking contest [6]. These videos are combinations of 8 video and 7 audio attacks referred as transformations. Video attacks are simulated camcording, insertion of pattern, reencoding, blurring, change of gamma, addition of noise, resizing, cropping, shifting, change of contrast, text insertion, flipping and frame dropping, or combinations of them. Audio attacks are MP3 compression, multiband companding, bandwith limiting, single-band companding, mixing with speech, multiband compressing and bandpass filtering, or combinations of them. Original videos have recorded at various frame rates within a wide range from 0.33 to 75.00 frames per second.

Figure 5 presents the key points that we would like to highlight in ITU MSPR CBCD system. In Fig.5(a), the ratio of the number of true positive matches over the number of total submitted matches, versus transformations is plotted for both BALANCED and NOFA profiles. Note that we have 56 audiovisual transformations generated as combinations of 8 video and 7 audio attacks. Similarly, Fig.5(b) illustrates

the ratio of the number of false alarms over the number of true positive matches, versus transformations. It can be concluded from Fig.5(a) and 5(b) that false alarm rate of our system is very small. Therefore the reported ratios in Fig.5(a) are high, namely, above 90% for NOFA, and above 80% for BALANCED profiles. On the other hand, the ratios plotted in Fig.5(b) remain within a small interval except a few transformation types. Since BALANCED and NOFA results are not too far from each other, we can say that it is possible to increase the performance for BALANCED profile by allowing more false alarms, which may cause an increase in miss count and total number of submitted queries. This can be performed by just adjusting the matching thresholds and making the runs from the scratch.

Figure 5 also shows that in some transformation types the performance is rather lower than others. The reason that we have low copy detection performance for some attack types, is because of the designed system does not robust to severe geometric attacks and picture-and-picture type queries. Our ongoing work is concentrated on increasing the developed system's robustness to severe geometric attacks and still scenes.



(a) (b)

Fig. 5: a) Ratio of true positives over number of total submitted queries versus transformatiion number. **b)** Ratio of false alarms over true positives versus transformation number.

4. Conclusions

The designed video content based copy detection system is robust to global attacks and subtle geometric attacks. Integrating the audio fingerprinting system obviously raised the matching performance. Using audio fingerprints improved the performance of matching significantly for picture-in-picture type queries, where video only queries return false results.

Currently we are working on reducing the computational complexity with sparse fingerprints while improving the robustness [7].

Acknowledgements

This research is supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) EEEAG under the project number 109E063.

References

[1] O. Gursoy, B. Gunsel, N. Sengor, "Transform Invariant Video Fingerprinting by NMF," in Proc. of CAIP 2009, pp. 452-459, 2009.

[2] S. Bucak, B. Gunsel, "Incremental subspace learning via non-negative matrix factorization," Pattern Recognition, vol. 42(5), pp. 788-797, 2009.

[3] J. Haitsma, T. Kalker, "A Highly robust audio fingerprinting system with an efficient search strategy," Journal of New Music Research, vol. 32(2), pp. 211-221, 2003.

[4] D. D. Lee and H.S. Seung, "Learning the parts of objects by nonnegative matrix factorization," Nature, vol. 401, pp. 788-791, 1999.

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

[6] A. F. Smeaton, P. Over, W. Kraaij, "Evaluation campaigns and TRECVid," in Proc. of the 8th ACM Int. Workshop on Multimedia Information Retrieval, MIR '06. ACM Press, New York, NY, pp. 321-330, Santa Barbara, California, USA, October 26 - 27, 2006, DOI=http://doi.acm.org/10.1145/1178677.1178722.

[7] O. Cirakman, B. Gunsel, N. Sengor, O. Gursoy, "Key-frame based video fingerprinting by NMF," Proc. IEEE ICIP, pp. 2373-2376, Hong Kong, October 2010.