A Fast and Robust Key-Frames based Video Copy Detection Using BSIF-RMI

Yassine Himeur, Karima Ait-Sadi and Abdelmalik Oumamne
Centre de Développement des Technologies Avancées (CDTA), Division TELECOM, Alger, Algerie

Keywords: Video Copy Detection, Key-Frames, Gradient Magnitude Similarity Deviation, Binarized Statistical Image Features, Relative Mean Intensity.

Abstract: Content Based Video Copy Detection (CBVCD) has gained a lot of scientific interest in recent years. One of the biggest causes of video duplicates is transformation. This paper addresses a fast video copy detection approach based on key-frames extraction which is robust to different transformations. In the proposed scheme, the key-frames of videos are first extracted based on Gradient Magnitude Similarity Deviation (GMSD). The descriptor used in the detection process is extracted using a fusion of Binarized Statistical Image Features (BSIF) and Relative Mean Intensity (RMI). Feature vectors are then reduced by Principal Component Analysis (PCA), which can more accelerate the detection process while keeping a good robustness against different transformations. The proposed framework is tested on the query and reference dataset of CBCD task of Muscle VCD 2007 and TRECVID 2009. Our results are compared with those obtained by other works in the literature. The proposed approach shows promising performances in terms of both robustness and time execution.

1 INTRODUCTION

CBVCD is proposed as an alternative or a complementary to the watermarking technology. It can detect copies without inserting any information and without altering the multimedia content **(**Lian et al., 2010). Unlike digital watermarking, Content based copy detection (CBCD) relies only on a similarity comparison of content between the original video and its various possible copies.

This technology is based on the fact that a media visually contains enough information for detecting copies. Therefore, the problem of CBCD is considered as video similarity detection by using the visual similarities of video clips.

Detection of a video copy in large video database is not an easy task because of the size of video data. By reducing the size of data that represents each video in the database, the video database manipulations such as indexing, copy detection and fingerprinting are accelerated. In fact, not all frames from a video sequence are equally important. A few informative frames that characterize the action for recognition are required. The reasons are; some video frames are irrelevant to the underlying activity, e.g. the frames with no action in them. They could be nuisance for the recognition. Also, the recognition speed can be greatly improved by using the informative key-frames without losing important information.to enable efficient representation and detection of digital video, many key-frames extraction techniques have been developed (Sujatha and Mudenagudi, 2011).

In this work, query and dataset video are systematically and efficiently reduced via a frame selection procedure which use GMSD (Xue et al., 2014) to detect key-frames in a video stream. Further refinement in the frame selection step is achieved using a robust feature representation based upon BSIF and the RMI of the selected subset of decoded frames. The procedure is presented in detail in the following sections.

The paper is organized as follows. We first describe related work in this field. Section 3 presents the main contribution of the paper including the key-frames extraction process and the feature extraction descriptor based on BSIF and RMI. We then present a fast video copy detection framework. In Section 4, we provide the effectiveness of the proposed approach based on the experimental evaluation and the comparison to other works. Finally, discussion and concluding remarks are given in Section 5.

Himeur Y., Ait-Sadi K. and Oumamne A..
A Fast and Robust Key-Frames based Video Copy Detection Using BSIF-RMI.
DOI: 10.5220/0005060000400047
In *Proceedings of the 11th International Conference on Signal Processing and Multimedia Applications* (SIGMAP-2014), pages 40-47
ISBN: 978-989-758-046-8

2 RELATED WORK

Most video copy detection algorithms based on global feature extract low-level feature from the video images to represent the video, but these algorithms are sensitive to various copy techniques, so the detection result is not satisfactory. In contrast to the global features, the local feature describes the structure and texture information of neighborhood of the interest point (Joly et al., 2007), having a good robustness generally to brightness, viewing angle, geometry and affine transformations. The techniques based on local feature are divided into five types: spatial methods, temporal methods, spatial-temporal methods, transform-domain methods and color methods.

On the other hand video copy detection approaches can be classified into two large groups. The first group includes non-key-frames based approaches which used the whole video sequence in the detection process. Jiang et al. (Jiang et al., 2013) proposed a rotation invariant VCD approach; each selected frame is partitioned into certain rings. Then Histogram of Gradients (HOG) and RMI are calculated as the original features. In (Cui et al., 2010), a fast CBCD approach based on the Slice Entropy Scattergraph (SES) is proposed. SES employs video spatio-temporal slices which can greatly decrease the storage and computational complexity. Yeh et al. (Yeh et al., 2009) proposed a frame-level descriptor for Large scale VCD. The descriptor encodes the internal structure of a video frame by computing the pair-wise correlations between geometrically pre-indexed blocks. In (Wu et al., 2009), Wu et al. introduced a Self-Similarity Matrix (SSM) based video copy detection scheme and a Visual Character-String (VCS) descriptor for SSM matching. Then in (Wu et al., 2009), the authors added a transformation recognition module and used a self-similarity matrix based near-duplicate video matching scheme. By detecting the type of transformations, the near-duplicates can be treated with the 'best' feature which is decided experimentally. In (Ren et al., 2012), Ren et al. proposed a compact video signature representation as time series for either global feature or local feature descriptors. It provides a fast signature matching through major incline-based alignment of time series.

The Second group contains key-frames based techniques. Zhang et al. (Zhang et al., 2010) proposed a CBVCD based on temporal features of key-frames. Chen et al. (Chen et al., 2011) introduced a new video copy detection method based on the combination of video Y and U spatiotemporal feature curves and key-frames. Tsai et al. (Tsai et al., 2009) developed a practical CBVCD After locating the visually similar key-frame, the methods of Vector Quantization (VQ) and Singular Value Decomposition (SVD) is applied to extract the spatial features of these frames. Then, the shot lengths are used as the temporal features for further matching to achieve a more accurate result. In (Chaisorn et al., 2010), Chaisorn et al. proposed framework composed of two levels of bitmap indexing. The first level groups videos (key-frames) into clusters and uses them as the first level index. The video in question need only be matched with those clusters, rather than the entire database. In (Kim et Nam, 2009), Kim et al. presented a method that uses key-frames with abrupt changes of luminance, then extracts spatio-temporal compact feature from key-frames. Comparing with the preregistered features stored in the video database, this approach distinguishes whether an uploaded video is illegally copied or not.

3 PROPOSED VIDEO COPY DETECTION MODEL

As aforementioned above most CBVCD system consist of three major modules: Key-frames extraction, Extraction of fingerprint (feature vector) and sequence matching. Fingerprint must fulfill the diverging criteria such as discriminating capability and robustness against various signal distortion. Sequence matching module bears the responsibility of devising the match strategy and verifying the test sequence with likely originals in the database. The architecture of our proposed CBVCD system is shown in Figure 1.

3.1 Key-Frames Extraction Process

In this paper, Key-frames extracted from each video shot are based on visual attention and structural similarity. The approach produces a gradient magnitude similarity maps from each frame. The similarity of the maps is then measured using a novel signal fidelity measurement, called Gradient Magnitude Similarity Deviation (Xue et al., 2014) . A frame will be chosen as key-frame if the value exceeds certain threshold.

GMSD is used to estimate global variation of gradient based local quality map for overall image quality prediction. It is proved in (Xue et al., 2014)

that the pixel-wise gradient magnitude similarity (GMS) between the reference and distorted images combined with a pooling strategy the standard deviation of the GMS map can predict accurately perceptual image quality and measure efficiently the distortion between original and distorted images.

The principle consist of convolving an image with a linear filter such as the classic Roberts, Sobel, Scharr and Prewitt filters and some task-specific ones. For simplicity of computation, the Prewitt filter is used to calculate the gradient among the 3×3 template gradient filters. Prewitt filters along horizontal (x) and vertical (y) directions are defined as (Xue et al., 2014):

$$h_x = \begin{bmatrix} \frac{1}{3} & 0 & -\frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} \\ \frac{1}{3} & 0 & -\frac{1}{3} \end{bmatrix}, \quad h_x = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & -\frac{1}{3} \\ 0 & 0 & 0 \\ \frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} \end{bmatrix}, \qquad (1)$$

Convolving h_x and h_y with the reference image r and distorted images d yields the horizontal and vertical gradient images. The gradient magnitude images of r and d at location i, denoted by $m_r(i)$ and $m_d(i)$ are computed by small local path in the original image r or d as follows:

$$m_r(i) = \sqrt{(r \otimes h_x)^2(i) + (r \otimes h_y)^2(i)} \qquad (2)$$

$$m_d(i) = \sqrt{(d \otimes h_x)^2(i) + (d \otimes h_y)^2(i)} \qquad (3)$$

where the symbol "⊗" denotes the convolution operation., The gradient magnitude similarity (GMS) map is computed as follows (Xue et al., 2014):

$$GMS(i) = \frac{2m_r(i)m_d(i) + c}{{m_r}^2(i) + {m_d}^2(i) + c} \qquad (4)$$

where c is a positive constant that supplies numerical stability. By applying average pooling to the GMS map, Gradient Magnitude Similarity Mean (GMSM) is obtained:

$$GMSM = \frac{1}{N}\sum_{i=1}^{N} GMS(i) \qquad (5)$$

where N is the total number of pixels in the image. A higher $GMSM$ score means a higher overall image quality. The standard deviations of the GMS map is computed, it is called Gradient Magnitude Similarity Deviation (GMSD):

$$GMSD = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (GMS(i) - GMSM)^2} \qquad (6)$$

Note that the value of GMSD reflects the range of distortion severities in an image. The higher the GMSD score, the larger the distortion range, and thus bigger the difference between two consecutive frames.

The proposed key-frames extraction is based on measuring the distortion between two consecutive frames for the whole video sequence to detect key-frames with significant change of the visual content. After calculating the GMSD difference between all the video frames sequence, a vector is obtained and each value of the vector is compared to a threshold. Only the fames with a distortion $dist$ that exceed the threshold value are considered as key-frames.

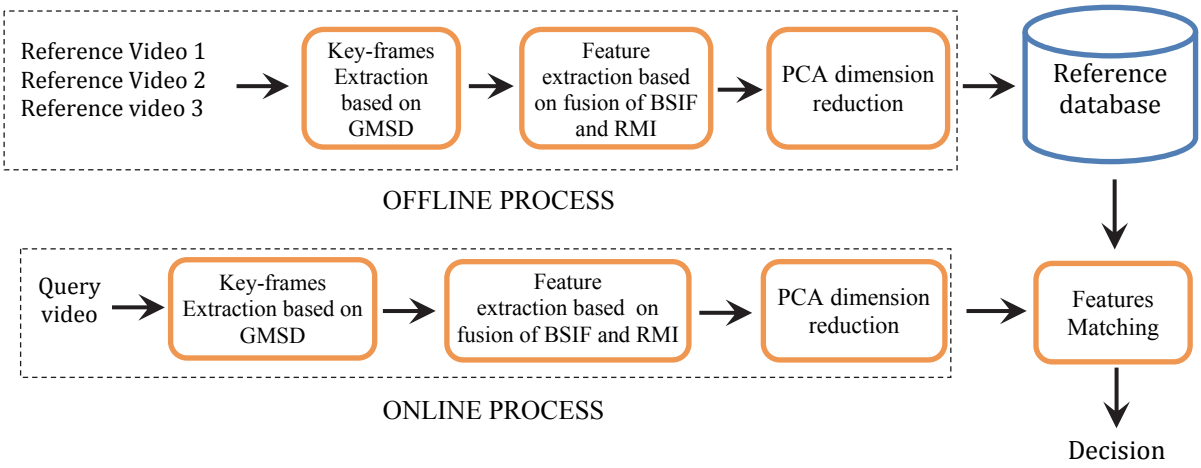

Figure 1: The architecture of the proposed CBVCD system.

$$dist_{i+1} = {F_r}^{i+1} - {F_r}^{i} \quad (7)$$

The threshold used in key-frames extraction process is computed using the following equation:

$$Thr = \alpha \times \frac{(\max_{GMSD} + \min_{GMSD})}{2} \quad (8)$$

where $0 < \alpha < 1$, $\max_{GMSD}$, $\min_{GMSD}$ are the maximum and minimum values obtained when computing the GMSD difference between consecutive video frames, respectively.

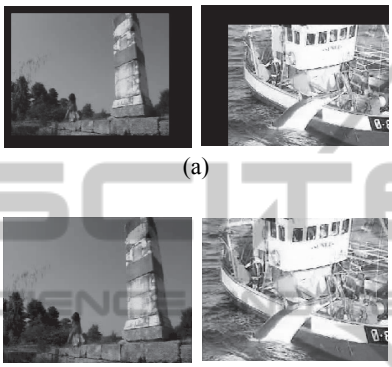

Figure 3: Examples of frames border removal. (a) original frames, (b) frames after border removal (samples frames from TRECVID 2009dataset) (Liu et al., 2009).

Remark 1: *The estimation of the key-frames is a delicate task; it must be carefully selected to minimize the video size and to be robust against different attacks. If key-frames extraction procedure is not robust against attacks, most of the key-frames will changed after applying an attack resulting in poor matching features results.*

To overcome this inconvenient, we must introduce some preprocessing on the transformed video frames, such as border removing. As shown in Figure 3 borders are removed by a simple method, which removes the first few lines of each direction (left, right, top, bottom) whose sum of intensity is less than a threshold (20% of the maximum in this paper).

3.2 Binarized Statistical Image Features

BSIF was recently proposed by Kannla and Rahtu for face recognition and texture classification (Kannala and Rahtu, 2012). It efficiently encodes texture information and is suitable for histogram based representation of image regions. To characterize the texture properties within each frame sub-region in a video sequence, the histograms of pixels BSIF code values are then used. A bit string is determined from a desired number of filters where each bit is associated with a different filter.

The set of filters is learnt from a training set of natural image patches by maximizing the statistical independence of the filter responses. Hence, statistical properties of natural image patches determine the descriptors. Given an image patch X of size $l \times l$ pixels and a linear filter W_i of the same size, the filter response is obtained by:

$$S_i = \sum_{u,v} W_i(u,v)X(u,v) = \mathrm{w_i}^{\mathrm{T}}\mathrm{x} \quad (9)$$

where w and x are vectors which represent the pixels of W_i and X. By setting $b_i = 1$ if $s_i > 0$ and $b_i = 0$ otherwise, the binarized feature b_i is obtained. For n linear filters W_i, a matrix W of size $n \times l^2$ is stacked and all responses at once, i.e., $\mathrm{s} = W.\mathrm{x}$ are computed. A bit string b is obtained by binarizing each element s_i of s as above. Thus, given the linear feature detectors W_i, computation of the bit string b is straightforward for more details; the reader can refer to (Kannala and Rahtu, 2012).

Independent component analysis are then used to learn the filters by maximizing the statistical independence of s_i in order to obtain a set of filters W_i. Additionally, the independence of s_i provides justification for the independent quantization of the elements of the response vectors. More details about the training set of image patches and how to obtain the filter matrix can be found in (Kannala and Rahtu, 2012).

3.3 Video Copy Detection

3.3.1 Feature Extraction Process

In this work, a new descriptor is introduced based on computing the BSIF characteristics of different images and their combination with the mean relative intensity of each region. It is extracted by encoding the pixel information of each key-frame.

For a given video, we segment it into K key-frames, which are the basic processing units in our approach. Each frame F_r in the key-frame selection is first converted to grayscale and resized to 128x128 pixels. A BSIF representation is then obtained after texture information encoding. We segmented the BSIF representation into n_B blocks to construct a BSIF histogram. Next, for the ith block

of a frame, the relative mean intensity (RMI) is calculated as follows:

$$RMI(i) = \sum_{p \in block(i)} p(x,y) / \sum_{p \in F_r} p(x,y) \quad (10)$$

where $p(x,y)$ represents the intensity of point (x,y). Figure 4 shows an example for $n_G = 4$. The connection between two nodes is determined by the content proximity between two blocks. We observe that two copies may not share common visual properties such as colors, textures, and edges; however, they often maintain a similar inter-block relationship.

As defined in Eq. (10), RMI is a global feature of each block. It represents the intra-block information and can help maintain a similar inter-block relationship between the query video and the reference. Besides, it is not sensitive to some complex brightness changes. Unlike previously reported work, which have focused on intensity/color variations only, the proposed algorithm adopt a combination of BSIF and RMI to describe the local distribution of each frame F_r.

BSIF is a new algorithm to face recognition used to provide local image features (Kannala and Rahtu, 2012). For each point of a block i, BSIF are calculated and BSIF histograms h of each block are constructed (Figure 4). As can be seen, if the query is flipped from the reference, the BSIF representation is opposite. To avoid this change, instead of directly using the BSIF representation, we divide their absolute values into certain number of bins. With the increasing of bins number n_B, the discriminative power of BSIF increases. However, the computation complexity also rises, and it will enlarge the influence of noise. To improve the discriminability, we combined BSIF and RMI features into the video description.

The features combination is used as weight; it is performed by combining the BSIF histograms to the corresponding RMI within a block as follows:

$$D_{nB\times 1} = BSIF_{nB\times nG} \times RMI_{nG\times 1} = \begin{pmatrix} BSIF_{1,1} & BSIF_{1,1} & BSIF_{1,nG} \\ BSIF_{2,1} & BSIF_{2,1} & BSIF_{1,nG} \\ \vdots & \vdots & \vdots \\ BSIF_{nB,1} & BSIF_{nB,1} \cdots & BSIF_{nB,nG} \end{pmatrix} \times \begin{pmatrix} RMI_1 \\ RMI_2 \\ \vdots \\ RMI_{nG} \end{pmatrix} \quad (11)$$

The final descriptor D encloses the intensity and local texture of each frame. From the calculating process, we find that D is overall a local descriptor with the length of n_B. It encodes the inner relationship of a frame and the local changes of intensities.

For a number n_{KF} of key-frames extracted from a query video, Key-frames descriptors are concatenated to form a matrix descriptor of size $n_{KF} * n_B$ which represents the matrix feature of a query video. The size of this matrix is then reduced using PCA to keep only a vector of size $4 \times n_B$. This means that each query video will be finally represented by a feature vector of $4 \times n_B$ variables.

Remark 2: *As copies may have different sizes with the original source. Here we employ a linear interpolation process to resize the query frames to the same size with its reference. This process is necessary because different sizes may cause different forms of the descriptors. Note also that each frame is first converted to grayscale before resizing.*

3.3.2 Matching Process

To match two descriptors D_1 and D_2, we choose Cosine distance as the similarity metric. In the matching process, a minimization process is employed. For an input query video, we find the clips with the minimal distance (maximal similarity) between descriptors in each source video. Then, we select the one with the lowest distance in the source. This distance is computed by the following equation

$$SIM = \cos(\theta) = \frac{D_1 . D_2}{\|D_1\| \|D_2\|} = \frac{\sum_{i=1}^{n} {D_i}^1 \times {D_i}^2}{\sqrt{\sum_{i=1}^{n} ({D_i}^1)^2} \times \sqrt{\sum_{i=1}^{n} ({D_i}^2)^2}} \quad (12)$$

4 EXPERIMENTAL RESULTS

4.1 Key-Frames Extraction Evaluation

A video summary should not contain too many key-frames since the aim of the summarization process is to allow users to quickly grasp the content of a video sequence. For this reason, we have also evaluated the compactness of the summary (compression ratio). The compression ratio is computed by dividing the number of key- frames in the summary by the length of video sequence.

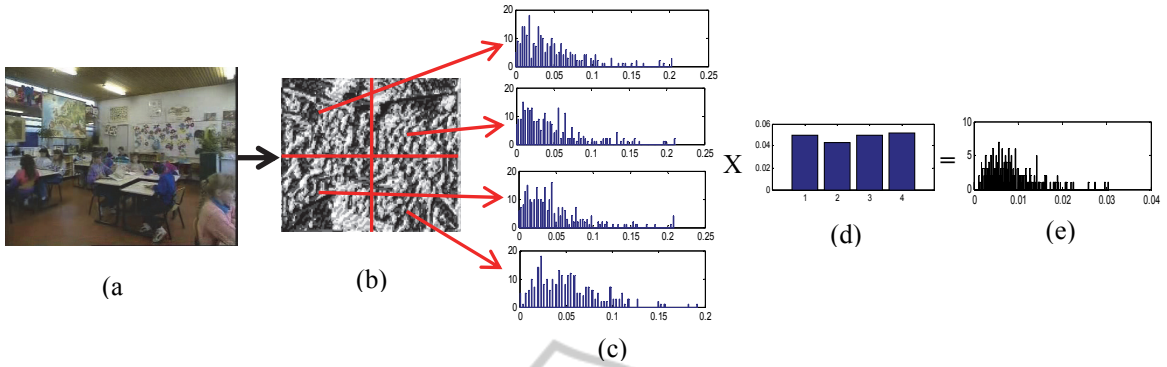

Figure 4: Illustrations of the developed descriptor with $n_G = 4$, $n_B = 256$. (a) original frame, b) BSIF representation with 4 grids , (c) BSIF histogram of each block; (d) RMI histogram, (e) fusion of BSIF and RMI descriptor.

For a given video sequence S_t, the compression rate is thus defined as:

$$CR_{(comp\ ratio)}(S_t) = 1 - \frac{\gamma_{NKF}}{\gamma_{NF}} \quad (13)$$

where γ_{NKF} is the number of key-frames in the summary, and γ_{NF} is the total number of frames in the video sequence. Ideally, a good summary produced by a key-frame extraction algorithm will present both high quality measure and a high compression ratio (i.e. small number of key-frames). Using the developed key-frames extraction algorithm we obtained an average compression ratio $CR = 99.6\ \%$. From the experiment, we can see that the representative key-frames can be extracted accurately and semantically from long video sequences or videos, objectively. Figure 5 shows the results of key-frames extraction with the proposed algorithm. The video depicts the process where there is an abrupt change in the video sequence.

The result illustrates that the algorithm is valid to segment the shot and extract the key-frames and it is of good feasibility and strong robustness.

Figure 5: Example of key-frames extracted from query video in TRECVID'09 database (TRECVID 2009).

In order to evaluate the performance of the key-frames extraction algorithm we first used TRECVID'09 (TRECVID 2009) definitions of detection recall and precision (DP and DR) as shown in Equations (14) and (18). Using the proposed algorithm we obtained an average recall of 97.7% and average precision of 100%.

We are currently building a larger ground truth data set for a more thorough evaluation of our algorithm.

$$DR = \frac{N_{FSDRT}}{N_{FRT}} \quad (14)$$

where N_{FSDRT} represents the number of frames shared between detected and reference transitions and N_{FRT} represents the number of frames of reference transitions

$$DP = \frac{N_{FSDRT}}{N_{FDT}} \quad (15)$$

where N_{FDT} represents the number of frames of detection transition.

4.2 Performance Evaluation Using MUSCLE VCD

In first time, we evaluate our approach on ten transformations. Experiments are conducted using the CIVR'07 Copy Detection Corpus (MUSCLE VCD) (Law-To et al., 2007). The corpus used is based on a task which consists to retrieve copies of whole long videos (ST1). The videos used in this corpus are size of 288×352 and come from web, TV archives and movies, and cover documentaries, movies, sport events, TV shows and cartoons. Meanwhile, there are 15 queries for with different

transformations like change of colors, blur, recording with an angle and inserting logos.

According to the evaluation plot (Law-To et al., 2007), criterion of the video copy detection scheme is defined as:

$$\text{Quality} = \frac{\text{NCorrect}_{\text{Num of Correct recognized}}}{\text{NTotal}_{\text{Num of Total Queries}}} \quad (16)$$

In our simulation, we set the BSIF parameter n_G to be 4/9/12/16 and n_B to be 256 bins. The threshold Thr is computed using $\alpha = 0.75$. Table 1 lists the matching qualities with the corresponding values. We can observe that the best results can be obtained by using a number of grids $n_G = 16$. Table1 and 2 show the result obtained using CIVR'07 corpus with the comparison to some existing approaches in the literature in term of time execution time and robustness to different attacks respectively. According to the table 3, the proposed approach runs faster than previous works.

Remark 3: *As n_G gets larger, the descriptor will possess more discriminate power. However, while the dimension increases, the descriptor is more sensitive to the border removal technique which may lower down the overall performance. In the other side, Theoretically, with more BSIF bins b_B, more information can be captured by the descriptor. However, for the existence of noise, the performance will be degraded if there are too many bins.*

4.3 Performance Evaluation Using TRECVID 2009 Dataset

The TRECVID 2009 dataset for CBCD (Liu et al., 2009) is also used to evaluate our approach.

Table 1: Matching qualities with different blocks (n_G).

n_G	Quality (%)
4	93
9	97
12	98.66
16	100

We prepare a corpus of 50 untransformed query videos and 350 transformed queries by applying seven classes of transformations to each query video. The different types of transformations are listed in Table 3. The queries last from 5 seconds to 2 minutes long.

Table 2: Transformation recognition qualities obtained with Muscle VCD 2007.

Transformation	Wu et al., 2009	Jiang et al., 2013	Proposed
AVC	89	93	**95.33**
Blur	88	100	**100**
Caption	95	100	**100**
Contrast	100	100	**100**
Crop	90	87	**93**
Mono	100	100	**100**
Noise	95	100	**100**
PicInPic	90	93	**96.66**
Ratio	100	100	**100**
Reduction	100	100	**100**

We use transformed queries to retrieve untransformed query videos from the derived dataset. The TRECVID 2009 dataset is challenging because the query videos are much shorter (i.e., 81 seconds long on average) and were produced by complicated transformations, such as, picture-in-picture and combination of various transformations.

5 CONCLUSION

In this paper, we propose fast and robust Key-frames for CBCD combining BSIF-RMI. RMI characterizes a global intensity level while BSIF represents the local distribution of the frame. The experiments obtained by the proposed approach show that the descriptor is effective and efficient. The matching time has been reduced efficiently by using only the key-frames and by reducing the dimension of feature vectors using PCA. Promising results are obtained for TRCVID 2009 and Muscle VCD 2007 databases in comparison to other works in the literature.

Table 3: Executive time of different approaches (in seconds).

Transformation	Time (sec)
Yeh and Cheng, 2009	1,394
Cui et al., 2010	849
Jiang et al., 2013	69
Proposed	**32**

Table 4: List of TRECVID'09 transformations and comparison of obtained recognition quality (TRECVID 2009).

T#	Transformation Description	Ren et al. 2012	Proposed
T2	Picture in picture	93	**96.66**
T3	Insertion of pattern.	100	**100**
T4	Strong re-encoding.	100	**100**
T5	Change of gamma	100	**100**
T6	Three random transf.: blur, gamma change, frame dropping, contrast, compression, ratio and noise	87	**93**
T8	Three random transf.: crop, shift, contrast, insert of pattern, vertical flip, picture in picture	100	**100**
T10	Combinations of 5 transformations chosen from T2 - T8	100	**100**

As a perspective, we will focus our work on the use of all the video sequence frames in the feature extraction task and how optimizing the execution time to deal with real time application.

REFERENCES

Chaisorn L., Sainui J. and Mander C., 2010. A Bitmap Indexing approach for Video Signature and Copy Detection. In *The 5Th IEEE Conf. on Industrial Electronics and Applications (ICIEA)*.

Chen X., Jia K. and Deng Z., 2011. An Effective Video Copy Detection Method. In *International Conference on Consumer Electronics, Communications and Networks (CECNet)*.

Cui P., Zhipeng W., Jiang S., Huang Q., 2010. Fast Copy Detection Based on Slice Entropy Scattergraph, In *IEEE Int. Conf. on Multimedia and Expo (ICME)*.

Jiang S., Su L. and Huang Q., 2013. Cui P., and Wu Z., A Rotation Invariant Descriptor for Robust Video Copy Detection. In *The Era of Interactive Media, pp 557-567.*

Joly. A, Buisson O, and Frelicot. C, 2007. Content-based copy detection using distortion-based probabilistic similarity search. In *IEEE Trans. on Multimedia.*

Kannala J. and Rahtu E., 2012. BSIF: Binarized Statistical Image Features. In *21st Int. Conf. on Pattern Recognition (ICPR)*.

Kim J. and Nam J. H., 2009. Content-based video copy detecion using spatio-temporal compact feature', In *11th Int. Conf. on Advanced Communication Technology, ICACT.*

Law-To J., Joly A., and Boujemaa N., 2007. Muscle-VCD-2007: a live benchmark for video copy detection, 2007. http://wwwrocq.inria.fr/imedia/civr-bench/.

Lian. S, Nikolaidis. N. and Sencar. H. T, 2010. Content-Based Video Copy Detection A Survey. In *Studies in Computational Intelligence, vol 282, pp. 253-273, Springer.*

TRECVID 2009, http://www-nlpir.nist.gov/project /tv2009 /tv2009.html

Ren J, Chang F. and Wood T., 2012. Efficient Video Copy Detection via Aligning Video Signature Time Series. In *Proceedings of the 2nd ACM International Conference on Multimedia Retrieval, No. 14.*

Roopalakshmi R. and Ram M. R. G., 2011. A Novel Approach to Video Copy Detection Using Audio Fingerprints and PCA. In *Procedia Computer Science, vol 5, 2011, pp. 149–156.*

Sujatha. C. and Mudenagudi. U., 2011. A Study on Keyframe Extraction Methods for Video Summary', In *International Conference on Computational intelligence and Communication Systems, 2011.*

Tsai C. C., Wu C. S., Wu C. Y. and Su P. C., 2009. Towards Efficient Copy Detection For digital Videos By Using Spatial and temporal Features. In *fifth Inter. Conf. on intelligent Information Hiding and multimedia signal Processing (IIH-MSP).*

Wu Z. P., Huang Q. M. and Jiang S. O., 2009. Robust copy Detection by Mining Temporal self-Similarities. In *IEEE Int. Conf. on Multimedia and Exp.*

Wu Z., Jiang S. and Huang Q., 2009. Near-Duplicate Video Matching with Transformation Recognition. In *Proc. Of the 17th ACM Int. Conf. on Multimedia, Pages 549-552.*

Xue W., Zhang L., Mou X. and Bovik A. C., 2014. Gradient Magnitude Similarity Deviation: A Highly Efficient Perceptual Image Quality Index. In *IEEE Trans. on Image Proc., vol 23(2), pp. 684-69.*

Yeh M. C., Cheng K. T., 2009. *A compact effective descriptor for video copy detection*. In *Proceedings of the 17th ACM international conference on Multimedia.*

Yeh. M. C. and Cheng K. T., 2009. Video copy detection by fast sequence matching. In *Proc. Of ACM Int. Conf. on Multimedia, pp. 633–636.*

Zhang Z., Zhang R. and Cao C., 2010. Video Copy Detection Based on Temporal Features of Key Frames', *In Int. Conf. on Art. Intelligence and Education (ICAIE).*