

Efficient Spatiotemporal Matching For Video Copy Detection in H.264/AVC Video

Mohammad Athar Ali
Department of Computer Engineering,
Aligarh Muslim University,
Aligarh, 202002, India

Eran A. Edirisinghe
Department of Computer Science,
Loughborough University,
Loughborough, LE11 3TU, UK

ABSTRACT

This paper proposes an efficient video copy detection method for the H.264/AVC standard. The mechanism is based on content based copy detection (CBCD). The proposed method divides each frame within a group of three consecutive frames into a grid. Each corresponding grid across these groups of frames is then sorted in an ordinal vector which describes both, the spatial as well as the temporal variation. This ordinal matrix based copy-detection scheme is effective in detecting not only a copied video clip but also its location within a longer video sequence. The technique has been designed to work in the compressed domain which makes it computationally very efficient. The proposed mechanism was tested on a number of video sequences containing copies which had undergone a variety of modifications. The results proved that the proposed technique is capable of detecting these copies effectively and efficiently and hence is suitable for forensic applications.

General Terms

Information Security, Pattern Recognition, Video Piracy

Keywords

Content-based copy detection, H.264/AVC, ordinal measurement

1. INTRODUCTION

The growing prevalence of distributing media (video, audio, images etc.) in a digital format also means that the media can be very easily duplicated/copied. Correspondingly, there is a growing need for stronger digital rights management (DRM) systems. DRM systems play an important role in protecting intellectual property rights (IPR) of digital multimedia content. There are a number of mechanisms that can be used to enforce DRM. They include methods like encryption, cryptographic hash algorithms, watermarking and Content Based Copy Detection (CBCD). All of these methods have their advantages and disadvantages. Some of the outstanding advantages of using CBCD can be easily found in technical literature.

CBCD methods use information that is unique to the digital media in order to detect copies. This information is usually termed as a 'signature' and is enough to identify that digital media. This implies that no additional information is required, in contrast to encryption and watermarking, in order to enforce DRM. The most popular statement in support of CBCD systems is that "the media itself is the watermark". Another added advantage is that the signature need not be extracted before the media is actually distributed. Finally,

since no additional processing is required, CBCD methods are usually computationally efficient which makes them suitable for applications where computing resources are limited or are at a premium.

CBCD methods have found acceptance in a number of applications such as detecting online copies of videos on torrents and in media tracking. Media tracking involves detecting the usage of a specific piece of media in terms of its time, location and frequency. This method has found widespread acceptance in marketing and advertising sectors especially for TV broadcasts, where a competitor's commercial can be tracked to obtain relevant information. The tracking results can be used for copyright management, claim unfair practices or royalty payments. Another interesting use of CBCD methods is to improve the search results of multimedia search engines where copies of the searched-for digital media can be removed before displaying the search results thereby reducing redundancy.

Signatures to perform CBCD can be extracted from spatial features such as luminance, from colour information such as the histogram, temporal features such as motion vectors, or even a combination of these features. Using spatial and colour features only for copy detection can be treated as a problem of image signature matching since these methods do not take into account the temporal nature of the video. Conversely, techniques developed using only temporal characteristics are ineffective against simple image processing operations such changes in luminance, contrast, colour etc. Thus for a CBCD method to be effective, a combination of spatial and temporal signatures are required. Keeping this requirement in mind, the approach presented in this paper uses a spatiotemporal methodology to detect copies of video sequences that have been encoded under the H.264/AVC standard.

The rest of the paper is organized as follows. Section 2 presents a brief overview of the state-of-the-art regarding CBCD systems. Section 3 explains the proposed copy-detection mechanism. Experimental results and performance analysis are presented in Section 4 followed by the Conclusion in Section 5.

2. EXISTING CBCD SYSTEMS

One of the earliest CBCD methods based on spatial features was proposed by Bhat and Nayar [1] wherein the ordinal measure of every frame in the clip was computed and then matched to detect copies. This was done by dividing each frame into $N \times N$ blocks and the average intensities of each block within that frame were sorted to give a rank matrix. Detecting copies can then be considered to be a problem of

matching the rank matrices frame-by-frame between the original and copied videos. Lee and Yoo [2] designed a video

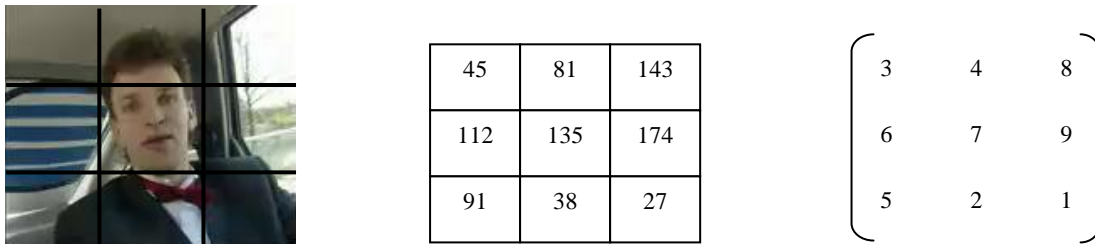


Fig 1: (a) Input Image divided into 3×3 equal sized blocks (b) Average intensity values in corresponding blocks (c) Ordinal Rank matrix derived from intensity values in (b)

fingerprinting method based on the centroid of gradient orientations. This method was claimed to be resistant to most common video processing steps such as resizing, compression, frame rate change etc. Other spatial techniques such as those based on differential luminance [3] and edge detection [4,5] have also been proposed.

Colour-based CBCD methods usually generate a unique signature from the colour histogram. Lienhart, Kuhmunch and Effelsberg [6] proposed a method where in the colour coherence vector was used to characterize key frames of a video clip. Sanchez, Binefa and Radeva [7] proposed the use of principal colour components within the histogram of key frames for copy detection.

A number of techniques based on the temporal nature of video sequences have also been proposed. Indyk, Iyengar and Shivkumar [8] proposed some of the earliest CBCD methods that exploited the temporal characteristics of video. They treated the time duration between shot transitions as a unique signature. Radhakrishnan and Bauer [9] used the frame difference method based on projections of difference images between consecutive video images to extract a robust signature. They claimed the method to be resistant towards signal processing operations such as changes in luminance, compression, resolution changes and scaling. Hampapur, Hyun and Bolle [10] designed a copy detection technique based on motion vectors.

However, experiments proved that ordinal signature based CBCD systems offer the most promising results. In fact, it has been proved in [10] that ordinal measurements not only offer the best performance when it comes to detecting copies but also high computational efficiency. Fig.1 shows a basic example of applying ordinal measurement to a single frame. The image is divided into $m \times n$ equal-sized blocks (usually $m = n$). This division makes the image independent of the size of the input frame. The average intensity in each block is calculated as shown in Fig.1(b). The average intensities are numbered in ascending/descending order as shown in Fig.1(c) giving rise to a rank matrix

It can be deduced that an overall increase/decrease in intensity levels will not change the order within the rank matrix and hence is resistant enough to detect a copy. Similarly, it has been proved that ordinal measurements are resistant to other commonly performed video processing steps such as histogram equalization, format conversion, transcoding etc.

Consequently, a number of techniques have been proposed over recent years that are based on the ordinal signature approach. Kim [11] proposed an ordinal measure of

DCT coefficients which was based on the relative ordering of AC magnitude values. This method offers better performance than ordinal measure of intensity which is unable to detect basic operations such as horizontal or vertical flipping of images. Kim and Vasudev [12] combined the spatial and temporal features of video to design a spatiotemporal sequence matching system. The system combined spatial matching of ordinal signatures and temporal matching of temporal signatures to detect copies. Chen and Stentiford [13] designed a CBCD system based on temporal ordinal measurements. Each frame was divided into a 2×2 grid and corresponding grids were sorted in an ordinal ranking sequence along a time series. This measurement captured both, the local and global description of temporal variation. Nie *et. al.*[14] partitioned the key frames into blocks and computed their ordinal measure. Then they evaluated its 64-point DCT and extracted a fingerprint out of it after discarding some of its components. This fingerprint was utilized to detect copies and it was claimed to be quite efficient in detecting copies in long video sequences.

Categorizing CBCD methods in terms of spatial, transform or compressed domain is another way of approaching the problem. Spatial [7, 15] and transform domain [11, 14] techniques though computationally more expensive, are simpler and more straightforward in design. Conversely, compressed domain approaches require a more in-depth understanding of the resulting bitstream in order to design the system but are usually more computationally efficient. A couple of compressed domain approaches based on motion vectors are reported in [10, 16] while another technique utilizing transform coded coefficients is presented in [17].

The idea proposed in this paper is to design a compressed domain CBCD method based on the spatiotemporal ordinal measurement. This would combine the high performance of spatiotemporal based ordinal measures with the computational efficiency of compressed domain approaches. The algorithm is designed based on features and characteristics that are unique to the H.264/AVC codec. The resulting CBCD system is computationally efficient and effective in detecting copies of videos that have been encoded under the H.264/AVC standard.

3. PROPOSED CBCD SYSTEM

It is clear that a combination of spatial and temporal information offers the most efficient way to match videos. To realize such a spatiotemporal system for matching H.264 videos, it is imperative to identify those aspects of the H.264/AVC encoder that generate the relevant information

and combine them together to generate unique signatures. Looking at the functioning of the encoder at different parameter settings [18], it can be noted that 4×4 *intrapredicted* macroblocks within the I-frames as well as *P*- and *B-skip* macroblocks within *P*- and *B*- frames respectively, are capable of retaining both, spatial and temporal information. We further argue that any modifications to the video such as sharpening or change in intensity value might change the number of 4×4 *intrapredicted*, *P*- and *B-skip* macroblocks in each of the respective frames but the ordinal rank matrix built out these frames would not be perturbed since the change would be consistent across these frames. Hence, they could be used as candidates to generate spatiotemporal signatures for a given video sequence.

3.1 Signature Design

Kim and Vasudev [12] proved that an ordinal matrix obtained by partitioning the frame into 2×2 regions is robust to most common modifications that can be done on a video sequence. In line with this argument, we begin by dividing the frames into 2×2 regions and representing them as TL (Top Left), TR (Top Right), BL (Bottom Left), BR (Bottom Right). We then count, in all the four regions, the number of *intra* 4×4 macroblocks within the I-frame, the number of *B-skip* macroblocks within the B-frame and the number of *P-skip* macroblocks within the P-frame. The resulting counts in each of the corresponding regions for each frame are then ranked as an ordinal vector along the time line. These vectors are then combined together to give the final ordinal matrix which acts as the signature. Fig.2 below explains signature generation using a combination of three different frame types.

Fig.2 (a) depicts the division of frames into 2×2 regions. In order to ensure that the macroblock count in each region of the frame is an integer, it may be necessary to divide the frame into unequal regions. However, we still attempt to divide the frame as equally as possible. For instance, if the frame resolution is 176×144 then there would be 11 macroblocks along the *x*-axis and 9 macroblocks along the *y*-axis as can be seen in Fig. 2(a). In this case, the division was made after the 5th macroblock along the *x*- and the *y*-axis,

starting from the top-left corner macroblock. In general, the approach adopted is:

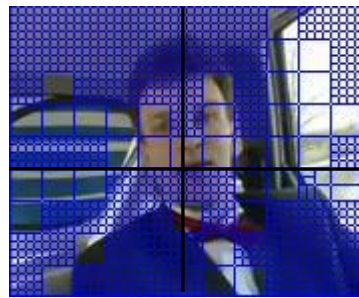
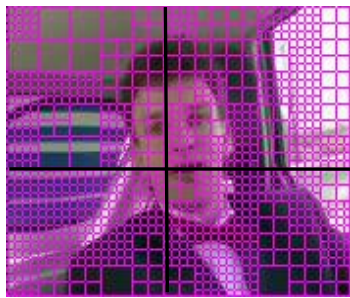
$$x_{partition} = \left\lfloor \frac{MBCount_x}{2} \right\rfloor, \quad y_{partition} = \left\lfloor \frac{MBCount_y}{2} \right\rfloor$$

where $x_{partition}$ is the macroblock position along the *x*-axis after which to draw the vertical partition line, $y_{partition}$ is the macroblock position along the *y*-axis after which to draw the horizontal partition line, $MBCount_x$ and $MBCount_y$ are the total number of macroblocks along the *x*-axis and *y*-axis respectively. The arrows in Fig.2(b) depict the macroblock count in the TL regions of the three frames being compared to generate the first row (vector) of the ordinal rank matrix. Similarly, the macroblock count from the remaining three regions can be used to realize the final ordinal matrix of Fig.2(c).

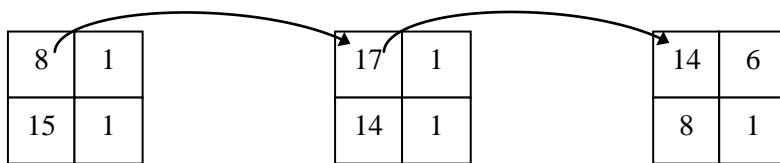
3.2 Matching Methodology

The matching process is performed according to the mechanism shown in Fig.3. The first three frames at the beginning of every group of picture (GOP) are utilized to build a part of the signature. These sub signatures are concatenated to constitute the complete signature of a given video sequence.

Figure 3(a) depicts the extraction of the signature from the query video. Similarly, signatures are extracted from every subsequence within the target video as shown in Fig.3(b). It should be noted that each subsequence within the target video has the same length as the query video except possibly the last (i.e. M'). Fig. 3(c) depicts the matching process wherein the query video signature is matched against target video signatures. The problem of detecting copies of a given query video involves detecting not only whether a copy exists within a target video but also its location. We can formalize the problem by defining a few notations and symbols as follows. Let the query video be denoted as $V_Q = \{v_q^0, v_q^1, v_q^2, \dots, v_q^{M-1}\}$ and the target video as $V_T = \{v_t^0, v_t^1, v_t^2, \dots, v_t^{N-1}\}$ with M and N being the total number of frames within the query and target videos respectively and where, $M \ll N$.



(a)



(b)

$$\begin{aligned} TL &= \begin{bmatrix} 3 & 1 & 2 \end{bmatrix} \\ TR &= \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \\ BL &= \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \\ BR &= \begin{bmatrix} 1 & 2 & 3 \end{bmatrix} \end{aligned}$$

(c)

Fig 2: Signature generation: (a) Frame division into 2x2 regions (b) Counting relevant macroblocks in each frame along a time line (c) Spatiotemporal ordinal matrix

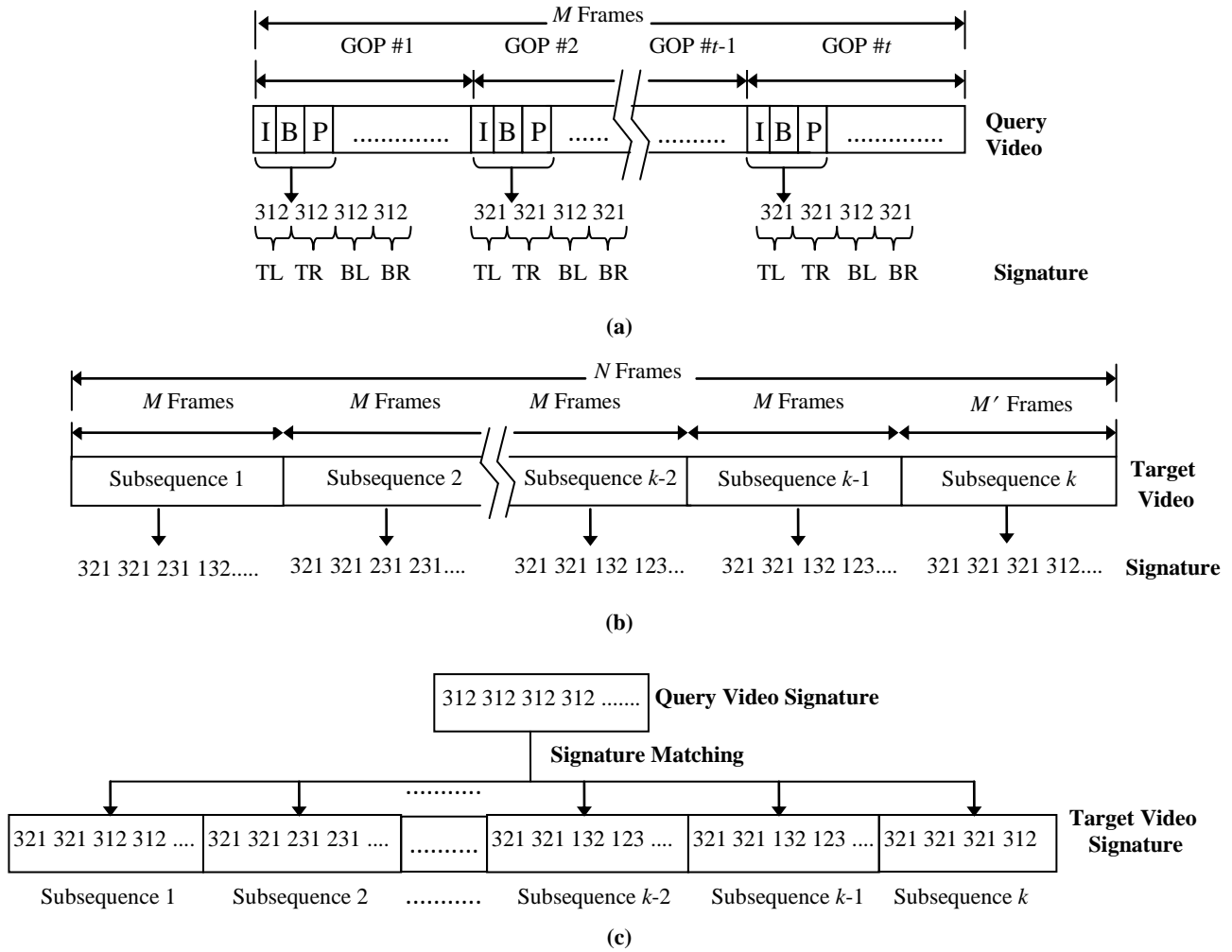


Fig 3: Signature generation and matching: (a) Extracting a part of the signature from the first three frames of every GOP within a sequence (b) Extracting signatures from each subsequence using the method in (a) (c) Matching query video to target video for detecting copies

A subsequence of m frames within V_T is defined as V_T^r where $r \in [r : r + m - 1]$, and with the value of r as $0 \leq r \leq N - m$ signifying the first frame of every subsequence. Further let each frame within V_T or V_Q be denoted as $V_i = \{V_i[0], V_i[1], \dots, V_i[P]\}$ with P being the total number of partitions within each frame. Then the ranking matrix of partition p within V_Q $= \{v_{q,p}^0, v_{q,p}^2, v_{q,p}^3, \dots, v_{q,p}^{M-1}\}$ can be denoted as $\pi_{q,p}$. The size of each $\pi_{q,p}$ would be of the order of $[1 \times m]$. Similarly, any subsequence within the target video $V_T^r = \{v_{t,p}^r, v_{t,p}^{r+1}, v_{t,p}^{r+2}, \dots, v_{t,p}^{r+m-1}\}$ can be denoted as $\pi_{t,p}^r$, also with a size of $[1 \times m]$. For the experimental results reported in this work, $m=M$ and the subsequences do not overlap i.e. to pick the next subsequence, r is not incremented by 1 but rather by gop_length . The above two assumptions lead to a set of non-overlapping subsequences of length M within V_T .

We can then define the problem of detecting copies as follows. Given a target video V_T , we say that a subsequence, V_T^r from V_T is a copy of V_Q if the distance $D(V_Q, V_T^r)$ is below a threshold $\epsilon \in [0, 1]$. The distance measure D is calculated as:

$$D(V_Q, V_T^r) = \frac{1}{P} \sum_{p=1}^P d(\pi_{q,p}, \pi_{t,p}^r)$$

and where d is calculated as:

$$d(\pi_{q,p}, \pi_{t,p}^r) = \frac{1}{C} \sum_{i=1}^M |\pi_{q,p}(i) - \pi_{t,p}^r(r+i-1)|$$

Each d is the normalized distance between two rank matrices. C is the maximum distance between two rank matrices π_k and π_j for all $(\pi_k, \pi_j) \in S_P$, with S_P being the set of all possible rank matrices with size P . C is obtained when the two rank permutations are reverse of each other. It is calculated as:

$$C = \sum_{i=1}^M |M+1 - (2 \times i)|$$

For the experiments reported in this paper, we used $P = 4$; hence $C = 8$. The copy detection mechanism proceeds as follows:

1. Initialize V_Q, V_T, V_T^r .
2. Initialize $r = 0, i = 0, gop_length = l$
3. Calculate $D(V_Q, V_T^r)$ and store in $D[i]$
4. Increase r by l, i by 1 and repeat step 3 until $r = N - m$
5. From $D[i]$ find minimum value D_{min} .
6. If $D_{min} < \epsilon$, declare the corresponding V_T^r as a copy and i as the location of the copy.

4. EXPERIMENTAL RESULTS

The proposed CBCD mechanism is implemented within the JM Reference Software [19] version 15.1. A requirement of using this software is that it accepts only raw YUV files as input. It was not possible to obtain a substantial database of YUV files to test the proposed system as there are only limited numbers of YUV QCIF (176×144) video sequences available online. A set of 24 different video sequences with different lengths were however obtained [20]. These video sequences covered almost all subjects like news, sports, scenery, architecture, interviews etc. These were concatenated to realize a longer target video sequence of 13,372 frames. Subsequences of length 30, 50, 100 and 150 frames were randomly selected from the above video sequence and 13 different transformations were applied to simulate copied video sequences and thus act as query videos. They are: increase brightness by 25%, decrease brightness by 25%, increase contrast by 25%, decrease contrast by 25%, decrease frame size down to 80%, increase frame size up to 120%, temporal smoothing, motion blurring, Gaussian radius-2 blurring, general convolution, decreasing frame rate down to 0.8 times the original rate, increasing frame rate upto 1.2 times the original rate and letter box. Query videos clips of varying lengths are then matched to the target video using the procedure outlined in the previous section.

The proposed algorithm is tested for varying values of ϵ , where ϵ is the normalized threshold value varying from 0 to 1. The receiver operating characteristics (ROC) curve was plotted as shown in Fig.4. Usually, the ROC curve is plotted as true positive rate (TPR) versus false positive rate (FPR). However, to present a comparison with the results presented in [12], Fig.4 depicts an ROC curve between FPR and false negative rate (FNR). These rates are calculated as follows. Let F_N be the number of false negatives i.e. number of copy-clips undetected and F_P be the number of false positives i.e. number of non-copy clips detected as a copy. Further, let N_T be the total number of non-copy clips and N_C be the total number of

copy clips. Then for a specific value of ϵ , the FPR and FNR can be calculated as:

$$FNR(\epsilon) = \frac{F_N}{N_C}, \quad FPR(\epsilon) = \frac{F_P}{N_T}$$

The ideal ROC curve would pass through the origin. This implies that closer the ROC curve passes by the origin, the better is the performance of the algorithm. As can be seen from Fig.4, the ROC curve for the proposed technique is closest to the origin when the video length is 50 frames. Interestingly, when the query video length is either decreased to 30 or increased to 100 frames and then further up to 150 frames, the ROC curve moves further away from the origin. Thus the query video length for the proposed CBCD system can be fixed at 50 frames to guarantee optimum performance. Fig.4 also indicates that the FPR and FNR rates for the proposed algorithm are better than those proposed in [12] and [21] when the query video length is 50 frames.

Another measure of the performance of any CBCD algorithm is to compute the precision and the recall rates for varying threshold values. These parameters are calculated as follows:

$$\text{Precision}(\epsilon) = \frac{\text{number of copy clips successfully detected leaving out false positives}}{\text{number of clips detected as a copy including false positives}}$$

$$\text{Recall}(\epsilon) = \frac{\text{number of copy clips successfully detected leaving out false positives}}{\text{total number of actual copy clips}}$$

Fig.5 shows the precision and recall rates plotted against normalized threshold values. The plot also includes the precision and recall rates reported in [12] and [21]. Kim and Vasudev [12] compared their method to the one proposed in [21] at a threshold value of $\epsilon = 0.1$. It can be seen from Fig.5 that at this threshold value, both the precision and recall rates for the proposed technique are higher than those reported in [12] and [21]. Even though the precision and recall rates are quite encouraging at a video query length of 30 frames, due to the poor ROC obtained (refer to Fig.4), we claim that optimum performance from the proposed system can be obtained at a query length of 50 frames and when $\epsilon = 0.1$.

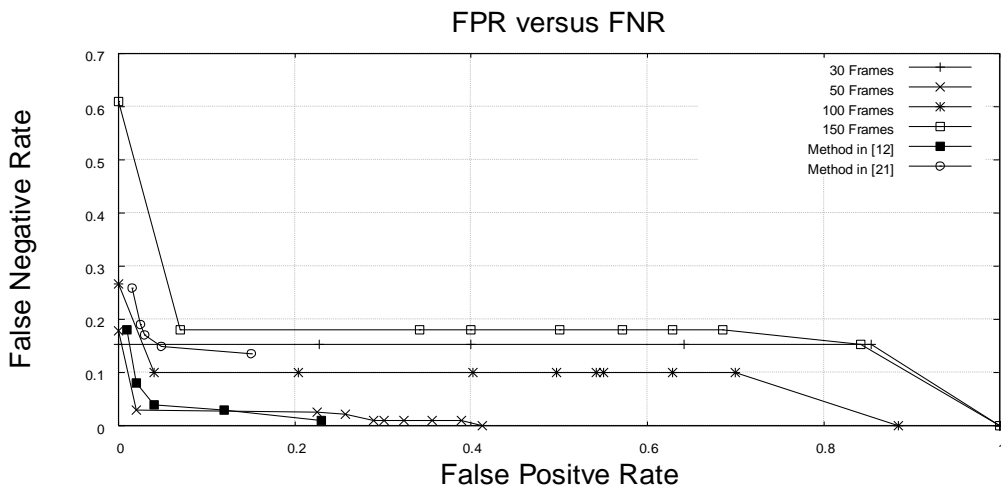


Fig. 4. Receiver operating characteristics (ROC) curve for FPR versus FNR

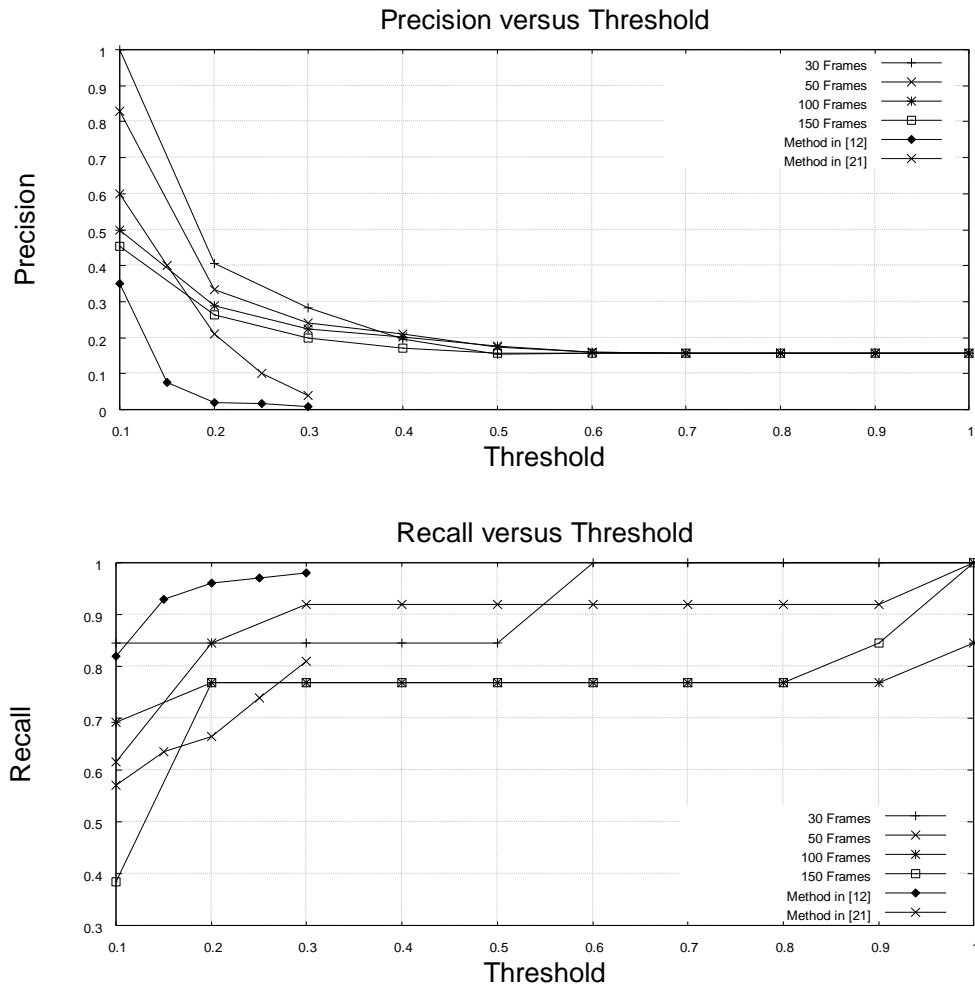


Fig 5: Precision and Recall rate versus normalized threshold

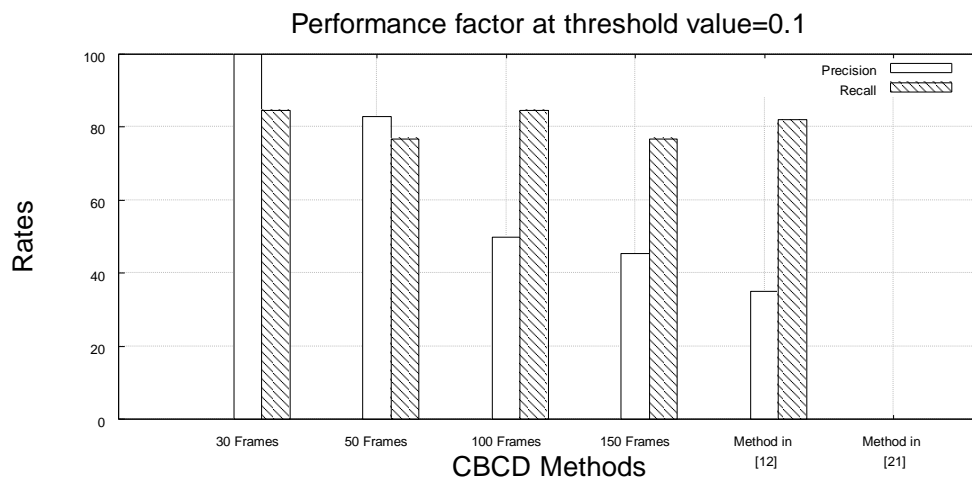


Fig 6: Precision and Recall rate versus normalized threshold

Fig.6 shows the precision and recall rates at different query lengths when compared to the rates reported in [12] and [21]. These values have again been obtained at $\varepsilon = 0.1$. As can be seen, the proposed method offers very encouraging results. Even though the recall rate is comparable to the method proposed in [12] however, the precision rate is much higher which signifies that there are less false detections within the

proposed system even with such short length query videos. The final evaluation of the proposed technique is in terms of the time taken to detect a copy. It is obvious that the proposed technique would be efficient since only the first 3 frames of each GOP within a sequence are used to generate the signature in contrast to every frame being used for signature generation and matching as was the case in [12]. In particular

only 3343 frames out of the above mentioned video sequence of 13,372 frames will play a role in the matching process. The copy detection time is computed from the time the signature is extracted out of the frames and matched. As per this condition, the time taken to match a 50 frame query video to the target video on a Pentium 4 PC, 3.4 GHz and 3GB of RAM was 0.38 seconds. The low detection time is also due to the fact that only partial decoding of the H.264/AVC bitstream is required in order to count the macroblock type thereby making it a compressed domain approach. In contrast, the methods reported in [12] and [21] are essentially spatial domain approaches. Finally, looking at the memory requirement, generating a three frame part-signature within each GOP and with each frame having 4 partitions gives a size of 12 bytes. With a GOP size of 12, a 50 frame query video sequence would have 5 GOP boundaries. This would give a total signature size of 60 bytes per video clip which is very low when considering memory and storage requirements.

5. CONCLUSION

This paper presents a compressed domain CBCD system designed to detect video copies encoded with the H.264/AVC standard. The proposed technique utilized a feature unique to the H.264/AVC standard wherein different regions of a frame and different types of frames are encoded as different macroblock types. This is done by reading the various macroblock types from a partially decoded H.264 bitstream. This method of signature extraction makes the proposed technique not only computationally efficient but also resistant to common video editing effects such as frame rate change/frame dropping, transcoding etc. In addition, use of ordinal methods to construct the signature guarantees that the proposed technique is resistant towards frame resizing, letter-box, sharpening and other common video processing steps thus making it suitable for forensic applications. Experimental results showed the technique to exhibit encouraging ROC behaviour. Further, the technique was able to not only detect modified copies but also their location in a longer video sequence within a very short span of time. Finally, since the proposed technique is a compressed domain method with a low processor and memory footprint.

6. REFERENCES

- [1] Bhat, D.N., and Nayar, S.K. 1998. Ordinal Measures For Image Correspondence. *IEEE Trans. Pattern Analysis and Machine Intelligence* (April 1998), 20 (4), 415-423.
- [2] Lee, S., and Yoo, C. D. 2008. Robust Video Fingerprinting For Content-Based Video Identification. *IEEE Trans. Circuits and Systems for Video Technology* (July 2008), 18(7), 983-988.
- [3] Oostveen, J., Kalker, T. and Haitsma, J. 2002. Feature Extraction and a Database Strategy for Video Fingerprinting. In *Proceedings of the 5th International Conference on Recent Advances in Visual Information Systems*. 117-128.
- [4] Iwamoto, K., Kasutani, E., and Yamada, A. 2006. Image Signature Robust to Caption Superimposition for Video Sequence Identification. In *Proceedings of the IEEE International Conference on Image Processing*. 3185-3188.
- [5] Hampapur, A., and Bolle, R. 2000. Feature Based Indexing for Media Tracking. In *Proceedings of the IEEE International Conference on Multimedia and Expo.3*. 1709-1712.
- [6] Lienhart, R., Kuhmunch, C., and Effelsberg, W. 1997. On the Detection and Recognition of Television Commercials. In *Proceedings of the IEEE International Conference on Multimedia Computing and Systems*. 509-516.
- [7] Sanchez, J.M., Binefa, X., and Radeva, P. 1999. Local color analysis for scene break detection applied to TV commercials recognition. In *Proceedings of Visual 99*, 237-244.
- [8] Indyk, P., Iyengar, G., and Shivakumar, N. 1999. Finding pirated video sequences on the Internet. Stanford Infolab Technical Report.
- [9] Radhakrishnan, R., and Bauer, C. 2007. Content-Based Video Signatures based on Projections of Difference Images. In *Proceedings of the IEEE 9th Workshop on Multimedia Signal Processing*. 341-344.
- [10] Hampapur, A., Hyun, K. and Bolle, R.M. 2002. Comparison of sequence matching techniques for video copy detection. In *Proceedings of SPIE, Storage and Retrieval for Media Databases*. 4676. 194-201.
- [11] Kim, C. 2003. Ordinal Measure of DCT Coefficients for Image Correspondence and its Application to Copy Detection. In *Proceedings of SPIE- IS&T Storage and Retrieval for Media Databases*. 199-210.
- [12] Kim, C., and Vasudev, B. 2005. Spatiotemporal sequence matching for efficient video copy detection. *IEEE Transactions on Circuits and Systems for Video Technology* (Jan.2005), 15(1), 127- 132.
- [13] Chen, L., and Stentiford, F. W. M. 2008. Video sequence matching based on temporal ordinal measurement. *Pattern Recognition Letters*. 29 (13). 1824-1831.
- [14] Nie, R., Ding, G., Wang, J., and Zhang, L. 2009. A New Fingerprint Sequences Matching Algorithm for Content-Based Copy Detection. In *Proceedings of the 5th International Conference on Information Assurance and Security*. 427-430.
- [15] Basharat, A., Zhai, Y., and Shah, M. 2006. Content based video matching using spatiotemporal volumes. *Computer Vision and Image Understanding* (Aug. 2006). 8 (4). 686-697.
- [16] Hampapur, A., and Bolle, R.M. 2001. Comparison of distance measures for video copy detection. In *Proceedings of the 2001 IEEE International Conference on Multimedia and Expo*. 737- 740.
- [17] Li, Z., and Chen, J. 2010. Efficient Compressed Domain Video Copy Detection. In *Proceedings of the 2010 International Conference on Management and Service Science (MASS)*. 1-4.
- [18] Wiegand, T., Sullivan, G.J., Bjontegaard, G., and Luthra, A. 2003. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology* (July 2003). 13(7). 560-576.
- [19] Reference JVT: Software Version 15.1
<http://iphome.hhi.de/suehring/tml/download/>
- [20] YUV video sequences: <http://trace.eas.asu.edu/yuv/>
- [21] Mohan, R. 1998. Video sequence matching. In *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing.6*. 3697-3700