

Fast Watermarking of MPEG-4 AVC/H.264 Encoded HDTV Video Bitstreams

Jakub Stankowski, Tomasz Grajek, Marek Domański
Chair of Multimedia Telecommunications and Microelectronics
Poznan University of Technology,
Poznan, Poland
{jstankowski, tgrajak}@multimedia.edu.pl, domanski@et.put.poznan.pl

Abstract— The paper describes watermarking of compressed video bitstreams. This technique is proven to be very fast and robust against various attacks including camcorder recording attack. The watermark is embedded using indirect modifications of transform coefficients in selected P- and B-frames. In the watermarked bitstream there exist sequences of watermarked pictures interleaved with sequences of unmodified pictures. The unmodified pictures serve as a reference for the watermark detection. The experimental results described for real-world MPEG-4 AVC/H.264 encoded HDTV video bitstreams are included.

Keywords- watermarking, compressed domain watermarking, HDTV, MPEG-4 AVC, H.264, camcorder recording attack.

I. INTRODUCTION

Great improvement achieved in high quality flat-panel display and digital video camera technologies allows users to record videos displayed on HDTV with little quality degradation. Distribution of illegal copies, made using such a method, may have a very significant economic influence on the “movie market”. Digital watermarking is one of the tools for protecting digital copyrights. This paper refers to watermarking for detection of illegal camcording of video content. Therefore, the watermarks should resist camcording, i.e. camcording attacks.

In literature one may find several studies dealing with camcorder attacks [4-7]. Unfortunately, most of these solutions are of high computational complexity mainly due to using uncompressed video content in a watermark embedder. Therefore, full decoding and re-encoding of video content is needed [7,8]. Moreover, in the vast majority of papers all experiments are conducted on low resolution video materials and achieved results cannot be directly referred to HDTV video materials.

In this paper, we propose a very fast watermarking method for MPEG-4 AVC/H.264 [1] bitstreams robust to very strong attacks, e.g. video camcorder recording. This method does not need full decoding and again re-encoding of the video content. Even full entropy decoding and re-encoding are not needed. Only selected headers of the selected pictures have to be modified, regardless of the used context-adaptive entropy method (CAVLC or CABAC) [2]. Watermarking is designed

to have minimal influence on video quality but, on the other hand, watermark has to be detectable after strong attacks including camcorder recording attack.

The main idea behind the proposed method is to embed a watermark into a compressed MPEG-4 AVC/H.264-compliant bitstream, without the need of full entropy de- and encoding of the bitstream. The proposal is to achieve this goal by computationally inexpensive indirect modifications of residual transform coefficients obtained by changing the quantization matrices. To be precise, only non-zero transform coefficients can be changed. Quantization matrix modification affects the whole image, there is no possibility to control the changes locally. This operation affects the values of residual signal in the decoder and thus modifies the reconstructed image. The idea is to change coefficients in the first row and the first column of the transform block.

As a matter of fact, the idea of watermarking by using quantization matrices has been already presented for MPEG-2 bitstreams [4,5]. However, as compared to the abovementioned methods, the proposed method provides an original approach to selection of the embedded coefficients and watermark detection.

II. PROPOSED METHOD

A. General idea

In MPEG-4 AVC/H.264 standard terminology, equivalent for the quantization matrix is scaling matrix. Therefore, in this article those terms will be used interchangeably.

The MPEG-4 AVC/H.264 standard defines default scaling matrices (quantization matrices) for luma and chroma components. Nevertheless, in the High profile, the standard provides a possibility to use custom quantization matrices for transform coefficients. Such matrices can be added to the bitstream in the sequence and/or picture parameter sets. A custom scaling matrix could be sent separately for intra and inter macroblocks and separately for luma and chroma components.

B. Profile issue

In MPEG-4 AVC/H.264, the quantization matrix (scaling matrix) transmission is possible in the High profile. In HDTV

broadcasts, sequences compliant with the High profile are mostly used. If the Main profile bitstream is to be watermarked, this bitstream has to be converted to the High profile. The set of compression tools of the High profile is a superset of those for the Main profile [3]. Therefore, every Main profile bitstream can be converted to the High profile by modifying the *profile_idc* in Sequence Parameter Set (SPS) and adding some flags required in the High profile. This modification does not have any effect on the reconstructed image but makes the stream decodable only by the High profile compatible decoder (standard HDTV decoder is compatible with the High profile of MPEG-4 AVC/H.264).

C. Watermark embedding

In MPEG-4 AVC/H.264, the syntax element containing scaling matrix is called scaling list [1]. In the Main profile bitstream (converted to the High profile), the watermark may be embedded only with use of 4x4 scaling matrices (only 4x4 transform size is available). In the High profile bitstream, the watermark can be embedded with use of both 4x4 and 8x8 scaling matrices in order to achieve sufficient watermark detectability.

For watermarking purposes, low-frequency AC coefficients were chosen, because we found that lower frequencies are more robust to attacks such as camcorder recording or transcoding. This observation is compliant with those already reported in the references [9]. Therefore, (0,1) and (1,0) coefficients have been selected for the 4x4 transform. These coefficients are not distorted by harmonics produced from other coefficients in non-linear systems that model the attacks. In 4x4 and 8x8 scaling matrices, the modified coefficients should correspond to similar spatial frequencies. Therefore, (0,2), (0,3), (2,0) and (3,0) coefficients have been selected for the 8x8 transform (see Fig. 1).

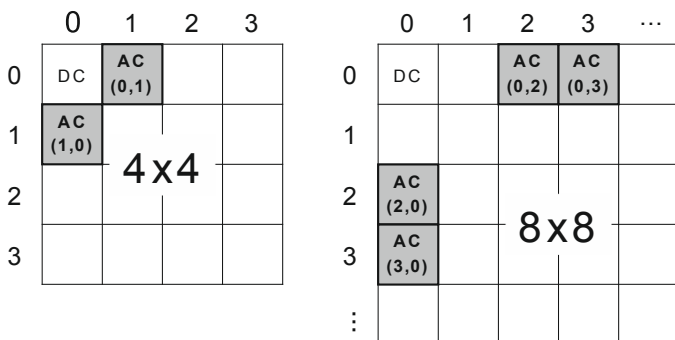


Figure 1. The modified transform coefficients, for 4x4 and 8x8 block size, respectively.

The watermark may be inserted into I-, P-, or B-frames. The scaling matrices for luma or/and chroma may be modified separately. After several experiments, we decided to embed the watermark only in the luma component in P- and B-frames. Watermarking of the chroma components cannot provide sufficient energy of watermark because the vast majority of transform coefficients are equal to zero. The main advantage of P-frame watermarking is a sufficient number of non-zero transform coefficients that can be modified in most of the macroblocks and limited subjective-quality image degradation.

In B-frames, there is only a small number of non-zero transform coefficients in most of the macroblocks, but watermarking of those frames provides better detection results without noticeable image quality degradation. Watermarking of I-frames by modification of scaling matrices results in significant image quality degradation mainly due to drift caused by intra prediction. It should be kept in mind that intra predicted macroblocks may be used by the encoder in all types of frames. Thus, the intra predicted blocks might be present in the watermarked sequences. The proposed method does not guarantee avoiding of intra prediction drift but it is possible to keep it unnoticeable by properly adjusting the watermark strength.

In MPEG-4 AVC/H.264, scaling matrix may be sent in Picture Parameter Set (PPS). In our proposal, modified scaling matrices are sent in additional PPSs. Once modified, the watermarked bitstream contains one original PPS and two additional PPSs. One of the latter contains a scaling matrix with increased vertical frequency coefficients (PPS-V) and the other contains a scaling matrix with increased horizontal frequency coefficients (PPS-H). This set of PPSs may be used to embed the watermark. Embedding process is basically a modification of slice headers in order to point out one of the three PPSs. For the unmodified slices the original PPS has to be indicated whereas for the watermarked slices one of the modified PPS (PPS-H or PPS-V) has to be chosen. Change of the PPS is done by *pic_parameter_set_id* field modification in slice header. Indicated PPSs will be used to decode the given slice, and modify appropriate coefficients. An example of the original and watermarked bitstream is shown in Fig.2.

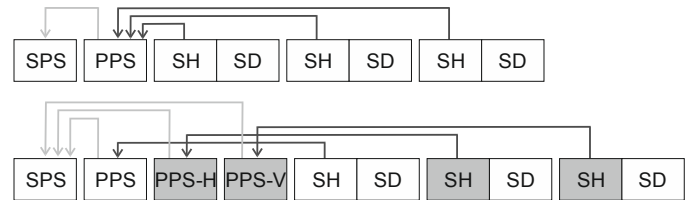


Figure 2. Bitstream modifications made during watermarking process: original bitstream (top), watermarked bitstream (bottom).

D. Embedding complexity and bitstream overhead

Because embedding is a simple slice header PPS swapping, the proposed watermarking technique is of very low computational complexity. Only PPSs and slice headers (SH) must be decoded, modified and encoded, but these units are very small (in tested bitstreams: SPS ~26 bytes, PPS ~5 bytes and SH ~5 bytes) and easy to encode. These units are encoded using fast Exp-Golomb code and may be processed without using CABAC or CAVLC [1]. Assuming structure of group of pictures (GOP) typical for broadcasting, PPS and SPS transmitting before every I-frame (to be precise – IDR-frame) and 30 fps, there is only about 360 bytes per second to tdecode. The computational requirements for the embedder are below 1 MIPS.

Bitstream overhead caused by the watermarking process is also very small. The additional PPS (with added scaling list syntax) has about 50 bytes, which means that about 200 bps

have to be added to the bitstream. For 10 Mbps bitstream, the watermarking overhead will be about 0.016%.

E. Watermark detection

Watermark detection is performed using the original and the investigated video sequence. Both sequences must be temporally aligned, but spatial alignment is not critical. Even temporal alignment does not need to be extremely accurate – shifts of 1-2 frames may be accepted if the sequences of watermarked frames are long enough. Temporal synchronization may be obtained using any of the known techniques.

During watermark embedding (0,2) and (0,3) or (2,0) and (3,0) coefficients have been modified (see section II.C). Therefore, the proposed detection algorithm uses transform coefficients energy to retrieve the watermark. The detector's input sequences (the original and the investigated) have to be analyzed in spectral domain. We decided to use the 8x8 transform as defined in MPEG-4 AVC/H.264. The averaged energy of each coefficient over 8x8 blocks in a frame is estimated as follows:

$$\overline{E}_{i,j} = \frac{1}{N} \sum_k (a_{i,j}^k)^2 \quad (1)$$

where $a_{i,j}$ is the amplitude of the (i,j)-th transform coefficient in k -th block, k is block number, N is the number of blocks in a single frame.

We assumed that the analyzed video sequence contains consecutive groups of watermarked and non-watermarked images. An example of image sequence from investigated video is shown in Fig. 3.

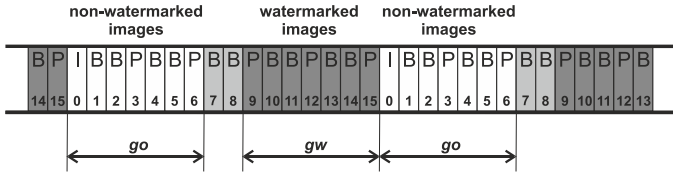


Figure 3. Example of investigated video sequence. Dark frames have been watermarked. Sequence contains non-watermarked groups of frames ($go1$ and $go2$) and watermarked groups of frames (gw).

In order to detect even a very slight watermark, strongly masked by marked data, content influence on the watermark has to be eliminated. In the proposed method it is assumed that transform coefficient energy distribution in transform coefficient being under investigation is quasi stable in consecutive frames. Therefore only modification of energy distribution is caused by watermarking. We compare transform coefficient energy distribution of investigated sequence with energy distribution of original sequence. First, the energy of coefficients is averaged separately for images from a watermarked group and for images from non-watermarked groups surrounding the watermarked group:

$$G_{i,j}^W = \sum_{l \in gw} \overline{E}_{i,j}^l, \quad (2)$$

$$G_{i,j}^O = \frac{1}{2} \left(\sum_{l \in go1} \overline{E}_{i,j}^l + \sum_{l \in go2} \overline{E}_{i,j}^l \right) \quad (3)$$

where: $G_{i,j}^W$ is an averaged energy for the watermarked images, $G_{i,j}^O$ is an averaged energy for non-watermarked images, l is the image number, i,j are coefficients' coordinates in transform block. Next, changes of the energy distribution between original (org) and investigated (cam) sequences are calculated as follows:

$$S_{i,j} = \frac{G_{i,j}^w(cam)}{G_{i,j}^o(cam)} - \frac{G_{i,j}^w(org)}{G_{i,j}^o(org)} \quad (4)$$

where: $S_{i,j}$ is a differential energy of the (i,j)-th coefficient.

Detector response bases on selected $S_{i,j}$ coefficients corresponding to the watermarked set of transform coefficients. As mentioned before, watermark was embedded by modifying (0,2) and (0,3) or (2,0) and (3,0) coefficients. Moreover, the camcorder recording causes that the watermark energy partially flows to (1,2), (1,3) or (2,1), (3,1) transform coefficients. Therefore, detector response has been defined as a weighted average of individual decision results:

$$r = a \cdot \text{sgn}(S_{0,2} - S_{2,0}) + b \cdot \text{sgn}(S_{0,3} - S_{3,0}) + c \cdot \text{sgn}(S_{1,2} - S_{2,1}) + d \cdot \text{sgn}(S_{1,3} - S_{3,1}) \quad (5)$$

where: r is detector response; $\text{sgn}(x)$ is sign function; a, b, c and d are weight matched to data statistics. Detector response $r < 0$ means that vertical frequency coefficients have been modified, $r > 0$ means that horizontal frequency coefficients have been modified.

III. EXPERIMENTAL RESULTS

A. Testing method

Tests have been performed on HDTV (1920x1080, 25 fps) video sequences of the length of 10 minutes each. Test sequences were encoded using x264 [10] encoder, configured to produce the MPEG-4 AVC/H.264 High-profile-compliant bitstreams. The prepared bitstreams had been watermarked and displayed on an LCD HDTV set. The displayed sequences were recorded by a HD camcorder. Afterwards, the recorded movie has been decoded and passed to the input of the watermark detector.

The watermark was embedded by changing the scaling matrix for P- and B-frames in the consecutive groups of pictures (GOP). For test purposes, two different random bit streams (A and B in Table 1) were used as the watermark payload. The watermark embedder was also set to mark the second half of GOP only (for example: when GOP length is equal to 16 only the last 3 P-frames and last 4 B-frames are modified – see Fig. 4). Each watermarked GOP corresponds to one bit from the watermark payload.



Figure 4. Example of watermarked GOP. Dark P- and B-frames are watermarked.

B. Perceptual quality

A watermark embedded using the proposed method is rather not visible. The most common effect would be seen as flickering spots related to positions where intra predicted macroblocks were used. These defects are hardly visible in still images but, when watermark is strong, they are visible and disturbing in video sequences.

Watermark visibility tests were performed and in most cases the watermark was rated as unnoticeable. Intra prediction related errors were evaluated weak and not frequent. In general, the watermark is not disturbing and does not stand out while watching test sequences.

C. Robustness

Watermark robustness tests were performed by recovering watermark payload from watermarked sequence. The effectiveness of watermark detection was calculated as a comparison of both: the detector response and embedded watermark data, in order to illustrate how many bits were detected correctly. Four different scenarios were tested to check robustness against most popular attacks:

- Scenario 1 (attack simulation): Low-pass filtering, 10% image cropping and 10% image resizing,
- Scenario 2: Camcording using HD camera,
- Scenario 3: Camcording using HD camera and downscaling to SD resolution,
- Scenario 4: Camcording using HD camera and framerate reduction (from 25 to 12,5 fps).

The effectiveness of watermark detection was collected in Table 1.

TABLE I. EXPERIMENTAL RESULTS

Sequence type	Payload bitstream	Properly detected bits [%] in scenario:			
		1.	2.	3.	4.
Soap opera 1	A	97.7	88.5	83.3	88.4
	B	97.4	87.6	82.0	87.8
Soap opera 2	A	98.4	88.5	88.5	88.5
	B	98.7	92.4	87.3	91.1
Historical novel	A	99.8	97.4	96.3	96.7
	B	99.9	99.6	98.7	99.2
Computer- animated film	A	96.6	87.2	82.7	84.7
	B	97.0	87.7	82.8	86.0
War film	A	94.9	76.2	72.4	73.8
	B	95.4	75.0	73.9	74.6
Average	A	97.5	87.6	84.6	86.4
	B	97.7	88.5	84.9	87.7

In Scenario 1, were combination of several attacks was simulated, more than 95% of bits was properly detected. For

scenarios, were real HD camcorder was used, properly detected bits ratio is not lower than 85%.

IV. CONCLUSIONS

The main advantage of the proposed method is very low computational complexity of the embedder. Therefore, it may be considered as a very fast watermarking solution. Additionally, increase of the watermarked bitstream size in comparison to original one is negligibly small. The results show that proposed method is robust against blurring, camcording, downscaling to SD resolution and downsampling to lower framerate. Our method is suitable for implementation on low processing power devices or systems, unlike other watermarking methods known from literature.

REFERENCES

- [1] ISO/IEC 14496-10: Inform. Tech. – Coding of Audio-Visual Objects – Part 10: Advanced Video Coding (2010).
- [2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, “Overview of the H.264/AVC video coding standard,” IEEE Transactions on Circuits and Systems for Video Technology, vol. 13, no. 7, pp. 560–576 (2003).
- [3] D. Marpe, S. Gordon, T. Wiegand, “H.264/MPEG4-AVC fidelity range extensions: tools, profiles, performance, and application areas”, IEEE International Conference on Image Processing, ICIP 2005, pp. 593 – 596, Genova, Italy, (2005).
- [4] M. Celik, J. Talstra, A.N. Lemma, S. Katzenbeisser, “Camcorder Capture Robust Low-Complexity Watermarking of MPEG-2 Bit-Streams”, IEEE International Conference on Image Processing, ICIP 2007, pp. 489-492, San Antonio, Texas, USA (2007).
- [5] D. Choi, H. Do, H. Choi, T. Kim, “A blind MPEG-2 video watermarking robust to camcorder recording”, Signal Processing, vol. 90, no. 4, pp. 1327-1332, Elsevier (2010).
- [6] Y. Wang, A. Pearmain, “Blind MPEG-2 video watermarking robust against geometric attacks: a set of approaches in DCT domain”, IEEE Transactions on Image Processing, vol. 15, no. 6, pp. 1536-1543 (2006).
- [7] H. Do, D. Choi, H. Choi, T. Kim, “Digital Video Watermarking Based on Histogram and Temporal Modulation and Robust to Camcorder Recording”, IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2008, pp. 330-335, Sarajevo, Bosnia & Herzegovina (2008).
- [8] F. Hartung, B. Girod, “Watermarking of Uncompressed and Compressed Video”, Signal Processing, vol. 66, no. 3 (Special issue on Watermarking), pp. 283-301, Elsevier (1998).
- [9] E. Nezhadarya, Z.J. Wang, R.K. Ward, “Watermark survival chance (wsc) concept for improving watermark robustness against JPEG compression”, IEEE International Conference on Image Processing, ICIP 2010, pp. 3697-3700, Hong Kong (2010).
- [10] Source code and description available on <http://x264.nl/>