Buletinul Ştiinţific al Universităţii "Politehnica" din Timişoara

Seria ELECTRONICĂ şi TELECOMUNICAŢII
TRANSACTIONS on ELECTRONICS and COMMUNICATIONS

Tom 53(67), Fascicola 2, 2008

High Capacity Digital Watermarking Algorithms for MPEG2 Compressed Video

Radu O. Preda[1], Dragoş N. Vizireanu[2]

Abstract – Video watermarking capacity is an evaluation of how much information can be hidden with in a digital video. In this paper we want to analyze the watermarking capacity for MPEG2 coded video using different blind watermarking schemes and introduce new watermarking algorithms with high watermarking capacity. The analyzed techniques are working in the spatial and DCT domain. We have tested the resistance of the watermarking algorithms against MPEG2 transcoding for different videos and improved the decoding BER using error correction codes.
Keywords: Digital Video Watermarking, Blind Detection, Spatial Embedding, Discrete Cosine Transform, MPEG-2 Compression.

I. INTRODUCTION

Digital watermarking for video is a fairly new area of research which basically benefits from the results for still images. Many algorithms have been proposed in the scientific literature and three major trends can be isolated. The most simple and straightforward approach is to consider a video as a succession of still images and to reuse an existing watermarking scheme for still images. Another point of view considers and exploits the additional temporal dimension in order to design new robust video watermarking algorithms. The last trend basically considers a video stream as some data compressed according to a specific video compression standard and the characteristics of such a standard can be used to obtain an efficient watermarking scheme. Each of those approaches has its pros and cons as detailed in Table I.

Table 1. Pros and cons of the different approaches for video watermarking

Adaptation image-video	Inherits all the results for still images	Computationally intensive
Temporal dimension	Video-driven algorithms which often permit higher robustness	Can be computationally intensive
Compression standard	Simple algorithms which make real-time achievable	Watermark may be inherently tied to the video format

A. *From still images to video watermarking*

In its very first years, digital watermarking has been extensively investigated for still images. Many interesting results and algorithms were found and when new areas, such as video, were researched, the basic concern was to try to reuse the previously found results. As a result, the watermarking community first considered the video as a succession of still images and adapted existing watermarking schemes for still images to the video.

Exactly the same phenomenon occurred when the coding community switched from image coding to video coding. The first proposed algorithm for video coding was indeed Moving JPEG (M-JPEG), which simply compresses each frame of the video with the image compression standard JPEG. The simplest way of extending a watermarking scheme for still images is to embed the same watermark in the frames of the video at a regular rate. On the detector side, the presence of the watermark is checked in every frame. If the video has been watermarked, a regular pulse should be observed in the response of the detector [1]. However, such a scheme has no payload. The detector only tells if a given watermark is present or not but it does not extract any hidden message. On the other hand, the host data is much larger in size than a single still image. Since one should be able to hide more bits in a larger host signal, high payload watermarks for video could be expected. This can be easily done by embedding an independent multi-bit watermark in each frame of the video [2].

[1] Electronics and Telecommunications Faculty, Department of Communications, Bd. Iuliu Maniu Nr. 1-3, 061071 Bucharest, e-mail radu@comm.pub.ro
[2] Electronics and Telecommunications Faculty, Department of Communications, Bd. Iuliu Maniu Nr. 1-3, 061071 Bucharest, e-mail nae@comm.pub.ro

B. *Integration of the temporal dimension*

The main drawback of considering a video as a succession of independent still images is that it does not satisfactorily take into account the new temporal dimension. The coding community has made a big step forward when they decided to incorporate the temporal dimension in their coding schemes and it is quite sure that it is the advantage of the watermarking community to investigate such a path. Many researchers have investigated how to reduce the visual impact of the watermark for still image by considering the properties of the Human Visual System (HVS) such as frequency masking, luminance masking and contrast masking. Such studies can be easily exported to video with a straightforward frame-per-frame adaptation. However, the obtained watermark is not optimal in terms of visibility since it does not consider the temporal sensitivity of the human eye.
Motion is indeed a very specific feature of the video and new video-driven perceptual measures need to be designed in order to be exploited in digital watermarking [3]. This simple example shows that the temporal dimension is a crucial point in video and that it should be taken into account to design efficient algorithms.

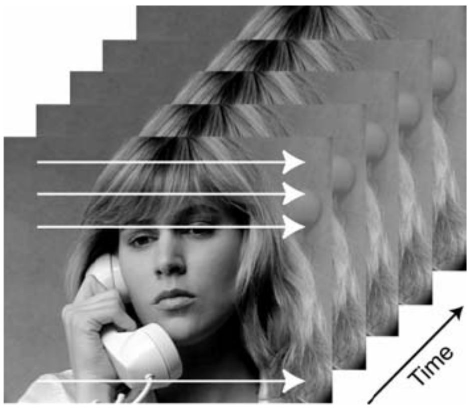

Fig. 1. Line scan of a video stream

C. *Exploiting the video compression standards*

The last trend considers the video data as some data compressed with a video specific compression standard. Indeed, most of the time, a video is stored in a compressed version in order to spare some storage space. As a result, watermarking methods have been designed, which embed the watermark directly into the compressed video stream. For example very specific parts of the video compression standard (run length coding) can be exploited in order to hide some information.
Watermarking in the compressed stream can be seen as a form of video editing in the compressed domain. Such editing is not trivial in practice and new issues are raised. The watermark can be directly inserted in the non-zero DCT coefficients of an MPEG video stream [4]. The first concern was to ensure that the watermarking embedding process would not increase the output bitrate. Nothing ensures indeed that a watermarked DCT-coefficient will be VLC-encoded with the same number of bits than when it was unwatermarked. A straightforward strategy consists then to watermark only the DCT coefficients which do not require more bits to be VLC encoded. The second issue was to prevent the introduced distortion with the watermark to propagate from one frame to another one. The MPEG standard relies indeed on motion prediction and any distortion is likely to be propagated to neighbor frames. Since the accumulation of such propagating signals may result in a poor quality video, a drift compensation signal can be added if necessary. In this case, motion compensation can be seen as a constraint.
However it could also be exploited so that the motion vectors of the MPEG stream carry the hidden watermark [5]. The components of the motion vector can be quantized according to a rule which depends on the bit to be hidden. For example, the horizontal component of a motion vector can be quantized to an even value if the bit to be hidden is equal to 0 and to an odd value otherwise.
All the frames of an MPEG coded video are not encoded in the same way. The intra-coded (I) frames are basically compressed with the JPEG image compression standard while the inter-coded (B and P) frames are predicted from other frames of the video. As a result, alternative watermarking strategies can be used depending on the type of the frame to be watermarked [6]. Embedding the watermark directly in the compressed video stream often allows real-time processing of the video. However the counterpart is that the watermark is inherently tied to a video compression standard and may not survive video format conversion.

II. WATERMARKING CAPACITY VERSUS EMBEDDING DOMAIN

The main embedding domains for digital video watermarking are the spatial domain, the transform domain (Discrete Cosine Transform, Discrete Fourier Transform, Discrete Wavelet Transform, Karhunen-Loewe transform) and the compressed domain.
The focus of this article is on analyzing how much watermark information can be inserted into MPEG-2 video without significant loss of quality. The techniques must be fit for real-time or near real-time processing. So we tried to maximize the watermarking capacity; that is why the methods are not very resilient to attacks. We have only studied the resilience of the proposed methods to the reduction of the MPEG-2 bitrate. From the point of view of watermarking capacity the spatial-domain and DCT-domain based techniques achieve the highest capacity. Methods working in other domains, like Discrete Fourier Transform, Discrete Wavelet Transform, Karhunen-Loewe Transform domain have to do a

transcoding from the DCT-domain into the other domain, which is usually very time consuming and would exceed the real-time criteria, so most of the works using these transform domains don't mention the processing time of the algorithms [7], [8]. On the other hand, the compressed domain techniques are very fast, but their capacity is quite low [9].

III. THE WATERMARKING SCHEMES

For determining the highest watermarking capacity we have tested different algorithms working in the spatial and DCT domain. The methods used are as simple as possible to minimize the processing time for watermark embedding and retrieval. In the following we present the techniques used for watermark embedding in one frame. The frame processing is discussed afterwards.

1. *Bitplane replacement in the spatial domain* – This is one of the easiest ways to embed a watermark. One watermark bit is embedded in the luminance value of every pixel and one in every 4 chrominance values. This is done by replacing the n-th Least Significant Bit of the luminance value with the watermark bit. For chrominance values, the mean of a 2x2 pixel block is calculated, the n-th Least Significant Bit of the mean is replaced with the watermark bit and the resulting chrominance value is copied to every location of the 2x2 block.

2. *Pixel quantization in the spatial domain* – Like the first method, one watermark bit is embedded in the luminance value of every pixel and one in every 4 chrominance values. The watermark bit is embedded in a luminance value by rounding the value to an even or odd quantization level. Rounding to an even quantization level embeds a zero, while rounding to an odd quantization level embeds a one.

3. *Block quantization in the spatial domain* - This method prevents MPEG-2 compression by embedding the same watermark bit in every 4x4 or 8x8 luminance block and in every 8x8 or 16x16 chrominance block respectively. The embedding is done in the same way as in the second method.

4. *Bitplane replacement in the DCT domain* – The method works in the same way as the first method, but in the DCT domain and only 22 DCT coefficients are used for watermark embedding (see Fig. 2). The 22 chosen coefficients are middle frequency DCT coefficients, because watermarking the low frequency DCT coefficient would greatly influence the visual quality and using the high frequency coefficients would make the algorithm fragile to MPEG-2 compression. The n-th Least Significant Bit of the DCT coefficient value is replaced with the watermark bit.

5. *Coefficient quantization in the DCT domain* - One watermark bit is embedded in every grey DCT coefficient from Fig. 2 for an 8x8 luminance block and one in every 4 DCT coefficients for an 8x8 chrominance block. The watermark bit is embedded in

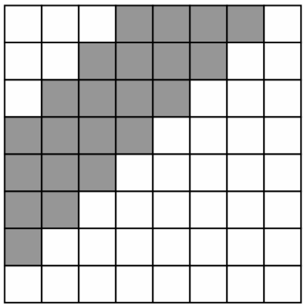

Fig. 2. Watermarked DCT coefficients for methods 4, 5 and 6

a DCT coefficient by rounding its value to an even or odd quantization level. Rounding to an even quantization level embeds a zero, while rounding to an odd quantization level embeds a one. This algorithm is faster than the same one in the spatial domain, because DCT recoding is not necessary for all frames.

6. *Block quantization in the DCT domain* - This method prevents MPEG-2 compression by embedding the same watermark bit in every DCT coefficient block for luminance values and in every 4 8x8 DCT coefficient blocks for chrominance values respectively.

Because the most common threat to MPEG-2 video watermarking is the MPEG-2 compression with different compression rates, we tried to protect the schemes by using two different methods. The first is adding an error correction code applied to the watermark bitstream. We have tested Hamming, cyclic and Reed-Solomon error correction.

The second protection method is the redundant embedding of the same watermark in consecutive frames.

IV. EXPERIMENTAL RESULTS

We performed the experiments on 10 different MPEG-2 coded videos of 10 seconds duration, resolution 720x576 and frame rate of 25 frames/s, having a GOP length of 12 frames (distance between consecutive I and P frames=3). The PSNR and Decoding Bit Error Rate (BER) shown in Table II are the mean values for the 10 videos. Parameter step is the quantization step used for methods 2, 3, 5 and 6.

For testing the resistance of our methods against MPEG-2 compression we compressed the watermarked videos at 4 and 1.5 Mbps and measured the decoding BER.

The error correction codes used were Hamming (15, 11) with codeword length of 15 bits and dataword length of 11 bits, which can detect and correct single-bit errors and Reed-Solomon with 8 bits/symbol, codeword length of 15 symbols and dataword length of 11 symbols.

The results show that the resulting PSNR values are in the range 32-36, which are acceptable values for video. We can see that the best decoding BER is achieved by the block quantization method in the DCT domain. The block quantization method in the spatial domain has promising results as well.

Table 2. Experimental results

Method	Error correction code	No. redundant frames	Step	Watermark size (kb/s)	6 Mbps (no compr.)		4 Mbps		1,5 Mbps	
					PSNR	Decoding BER (%)	PSNR	Decoding BER (%)	PSNR	Decoding BER (%)
1.	NO	1		15187	35,88	0,7404	33,37	34,6802	33,24	39,5503
		12		1266	35,82	0,0000	33,35	30,2663	33,22	34,2657
	Hamming	1		11137	35,86	0,0000	33,34	26,2645	33,26	30,0487
		12		928	35,79	0,0000	33,38	25,0225	33,25	28,6872
	Reed - Solomon	1		11137	35,78	0,0000	33,36	28,3068	33,22	32,2548
		12		928	35,87	0,0000	33,35	24,6580	33,24	27,5793
2.	NO	1	4	15187	35,12	0,0570	32,87	28,7502	32,76	31,6586
			8	15187	33,56	0,0000	31,90	25,3697	31,80	28,3302
		12	4	1266	35,21	0,0000	32,88	26,0245	32,77	28,3684
			8	1266	33,43	0,0000	31,92	22,8563	31,83	24,3050
	Hamming	1	4	11137	35,18	0,0000	32,85	22,5688	32,74	25,3695
			8	11137	33,31	0,0000	31,89	20,1124	31,78	23,2550
		12	4	928	35,23	0,0000	32,87	20,3698	32,78	22,6588
			8	928	33,47	0,0000	31,91	18,6589	31,83	20,1020
	Reed - Solomon	1	4	11137	35,16	0,0000	32,88	21,5690	32,78	24,0125
			8	11137	33,59	0,0000	31,90	20,6656	31,81	23,2355
		12	4	928	35,19	0,0000	32,87	18,5633	32,77	20,5688
			8	928	33,47	0,0000	31,91	17,0245	31,83	19,3600
3.	NO	1	4	237	35,24	0,0000	32,86	1,4004	32,75	2,7023
			8	237	33,61	0,0000	31,91	0,0024	31,81	0,0821
		12	4	20	35,13	0,0000	32,89	0,1296	32,76	0,3302
			8	20	33,48	0,0000	31,94	0,0000	31,84	0,0350
	Hamming	1	4	174	35,17	0,0000	32,83	0,4654	32,71	0,9036
			8	174	33,29	0,0000	31,87	0,0012	31,77	0,0043
		12	4	15	35,2	0,0000	32,88	0,0063	32,78	0,0234
			8	15	33,4	0,0000	31,93	0,0000	31,84	0,0000
	Reed - Solomon	1	4	116	35,15	0,0000	32,86	1,1254	32,76	2,4020
			8	116	33,53	0,0000	31,91	0,0000	31,82	0,0013
		12	4	10	35,12	0,0000	32,88	0,0000	32,77	0,0000
			8	10	33,46	0,0000	31,89	0,0000	31,81	0,0000
4.	NO	1		5221	35,92	0,0000	33,39	31,3658	33,25	35,5500
		12		435	35,89	0,0000	33,36	27,3658	33,23	30,1203
	Hamming	1		3828	35,92	0,0000	33,37	24,2360	33,28	28,3354
		12		319	35,87	0,0000	33,40	20,9640	33,26	23,2201
	Reed - Solomon	1		3828	35,86	0,0000	33,37	25,3691	33,24	28,4402
		12		319	35,87	0,0000	33,39	19,0230	33,24	22,0233
5.	NO	1	4	5221	35,22	0,0000	32,89	22,3658	32,78	25,2365
			8	5221	33,63	0,0000	31,92	17,5680	31,82	30,2537
		12	4	435	35,23	0,0000	32,91	20,3658	32,78	22,3555
			8	435	33,48	0,0000	31,93	15,2365	31,82	17,3685
	Hamming	1	4	3828	35,21	0,0000	32,87	16,2300	32,76	19,2500
			8	3828	33,42	0,0000	31,92	12,0352	31,78	15,3620
		12	4	319	35,26	0,0000	32,90	13,2560	32,78	15,3658
			8	319	33,47	0,0000	31,93	10,5362	31,86	12,7821
	Reed - Solomon	1	4	3828	35,19	0,0000	32,91	17,0253	32,78	20,1579
			8	3828	33,62	0,0000	31,91	11,7895	31,83	14,2304
		12	4	319	35,2	0,0000	32,89	13,0258	32,79	15,2387
			8	319	33,5	0,0000	31,92	9,8456	31,85	11,0245
6.	NO	1	4	237	35,26	0,0000	32,88	0,8562	32,75	1,5426
			8	237	33,65	0,0000	31,92	0,0000	31,82	0,0121
		12	4	20	35,18	0,0000	32,91	0,0596	32,77	0,1203
			8	20	33,52	0,0000	31,95	0,0000	31,86	0,0068
	Hamming	1	4	174	35,2	0,0000	32,86	0,1025	32,73	0,3036
			8	174	33,33	0,0000	31,89	0,0000	31,77	0,0009
		12	4	15	35,24	0,0000	32,91	0,0000	32,79	0,0028
			8	15	33,45	0,0000	31,95	0,0000	31,85	0,0000
	Reed - Solomon	1	4	174	35,18	0,0000	32,86	0,1425	32,78	0,2783
			8	174	33,58	0,0000	31,93	0,0000	31,84	0,0000
		12	4	15	35,14	0,0000	32,89	0,0000	32,79	0,0000
			8	15	33,49	0,0000	31,92	0,0000	31,83	0,0000

These two methods are the most resilient against MPEG-2 compression as well, because they insert watermarks with spatial and temporal redundancy. The error correction codes improve the overall BER as well, with an acceptable loss of capacity.

REFERENCES

[1] M. Barni, F. Bartolini, R. Caldelli, A. De Rosa, and A. Piva, “A Robust Watermarking Approach for Raw Video”, *Proceedings of the Tenth International Packet Video Workshop*, 2000.

[2] J. Dittmann, M. Stabenau, and R. Steinmetz, “Robust MPEG Video Watermarking Technologies”, *Proceedings of ACM Multimedia*, pp. 71-80, 1998.

[3] S.-W. Kim, et al., “Perceptually Tuned Robust Watermarking Scheme for Digital Video Using Motion entropy Masking”, *Proceedings of the IEEE International Conference on Consumer Electronics*, pp. 104-105, 1999.

[4] F. Hartung and B. Girod, “Watermarking of Uncompressed and Compressed Video”, *Signal Processing*, 66(3), pp. 283-301, 1998.

[5] F. Jordan, M. Kutter, and T. Ebrahimi, “Proposal of Watermarking Technique for Hiding/Retrieving Data in Compressed and Decompressed Video”, *ISO/IEC JTC1/SC29/WG11*, 1997.

[6] C.T. Hsu and J.-L. Wu, “DCT-based Watermarking for Video”, IEEE Transactions on Consumer Electronics, 44(1):206-216, 1998.

[7] Liu, H. M., N. Chen, J. W. Huang, X. L. Huang and Y. Q. Shi, “A robust DWT-based video watermarking algorithm”, *Proc. of IEEE International Symposium on Circuits and Systems*, vol.3, pp.631-634, 2002.

[8] D. Simitopoulos, S. A. Tsaftaris, N. V. Boulgouris, and M. G. Strintzis, “Digital watermarking of MPEG-1 and MPEG-2 multiplexed streams for copyright protection,” *Proc. Second International Workshop on Digital and Computational Video*, Tampa, FL, USA, Feb. 2001, pp. 140-147.

[9] Z. Zhao, N. Yu, X. Li, “A novel video watermarking scheme in compression domain based on fast motion estimation”, *International Conference on Communication Technology Proceedings*, vol. 2, 9-11 April 2003, pp. 1878 – 1882