CAMCORDER CAPTURE ROBUST LOW-COMPLEXITY WATERMARKING OF MPEG-2 BIT-STREAMS

Mehmet Celik, Joop Talstra, Aweke Lemma and Stefan Katzenbeisser

Information and System Security Group
Philips Research Europe
5656AE, Eindhoven, The Netherlands

ABSTRACT

Unauthorized re-distribution remains a significant threat for emerging electronic movie distribution services. In this paper, we propose a forensic tracking watermark for MPEG-2 bit-streams that can be employed to complement Digital Rights Management and conditional access systems in electronic movie distribution. The watermark is embedded by modulating a subset of quantization matrix entries, which are periodically present in the MPEG-2 headers. When observed over time the watermark can be detected even after cropping, de-interlacing, resizing and DivX compression at 300 kbps or after being captured with a video camera from a flat-screen TV. As the method modifies only a small part of the bit-stream ($\approx$ 100 bytes per second), it can be readily implemented in resource constrained environments like the current generation set-top boxes, without costly hardware upgrades.

***Index Terms*—** copy protection, forensic tracking, fingerprinting, MPEG-2

1. INTRODUCTION

In the last decade, movie distribution revenues have been shifting from theatrical releases to home videos (e.g. DVD sales). In the next decade, we expect this trend to continue with more emphasis on electronic distribution channels such as video-on-demand or download-to-buy services over cable networks or the Internet. One factor facilitating the growth of the electronic content distributing market is a shorter release time, i.e. the time it takes a movie to be available for download after its theatrical release. Another is the “long tail” phenomenon whereby value can be extracted from a large catalogue of relatively unpopular titles. Despite the apparent shift, unauthorized re-distribution of movies without compensation of rights holders remains a threat. Increasing availability of broadband connections and tools for sharing video content exacerbates the problem further.

The first line of defense against unauthorized re-distribution of movies delivered to the consumer via electronic distribution means is encryption. The content is encrypted before it is transmitted to the consumer, for instance at the cable head-end. Often, decryption only takes places on a dedicated set-top box (STB) right before decompression. Whenever possible, the connection between the STB and the TV set is further protected by link-encryption.

The protection offered by encryption, however, does not extend beyond the digital realm. Displayed content can be captured when it is output using analog connections or rendered on the TV screen. Analog connections are being rapidly replaced by higher quality digital connections (with cryptographic protection) and may be phased-out in the future. On the other hand, video rendering on the TV screen will always be necessary. Even today it is possible to copy content with a reasonable quality by displaying it on an LCD screen and capturing it with a digital video camera. We call this scenario the *camcorder capture attack*. Moreover, the advances in flat-panel display and video camera technologies will enable less expensive and higher quality copies in the future.

Forensic tracking watermarking is a complementary technology where each authorized copy of the content is watermarked with a unique identifier that links the content to the consumer who acquires it. Upon discovery of a content copy being distributed without authorization, e.g. on a peer-to-peer network, the copy is probed for the presence of a watermark, which reveals the identity of the consumer who has distributed the content without authorization. As legal measures follow, the watermark acts as a deterrent against unauthorized re-distribution. Forensic watermarking complements encryption by “encouraging” good behavior even after the content is rendered in the analog domain.

When set-top boxes are used as the point for embedding forensic watermarks, the complexity of the watermark embedding process gains importance. Low-complexity methods which preferably embed the watermark directly into the compressed bit-stream are desirable.

MPEG-1, MPEG-2 [1] and MPEG-4 (Part 2) [2] video compression standards are based on hybrid coding. A video sequence is divided into groups-of-pictures (GOPs). The first picture (I-picture) is intra-coded by applying an 8×8 block DCT transform, quantizing DCT coefficients according to a quantization matrix, zig-zag scanning, run-length and entropy coding. The quantization matrix indicates one quantization level for each frequency coefficient. Due to the relatively low sensitivity of the human visual system to high frequency quantization errors, coarser quantization is applied to higher frequencies. The remaining pictures in the GOP are inter-coded. In the motion estimation step, each block of pixels is predicted from the preceding picture (for P-picture) or from both the preceding and subsequent pictures (for B-picture). The prediction error is compressed in a manner similar to the I-pictures. Motion vectors are separately entropy coded and multiplexed into the bit-stream.

In the literature various methods have been proposed for watermarking of MPEG bit-streams. In [3], Hartung describes an algorithm which parses the bit-stream, entropy and run-length decodes quantized DCT coefficients (both for intra-coded pixel blocks and inter-coded prediction error blocks) and adds a 2-dimensional wa-

1-4244-1437-7/07/$20.00 ©2007 IEEE

termark pattern in the transform domain. Modified DCT coefficients are entropy coded and re-packed to form the watermarked bit-stream. In [4], Langelaar et al. propose an alternate approach where the watermark is embedded by selectively eliminating small quantized DCT coefficients according to a watermark pattern. This approach reduces the complexity of the earlier method as it can be implemented as merging of two VLC (variable length coding) codewords. Whereas both methods are significantly faster than fully decoding the video, embedding the watermark and re-encoding the video, they still require considerable computational resources. Moreover, due to their spatial nature, recovering the watermark after camcorder capture, which introduces geometric deformations, becomes a major challenge. On the other hand, two other algorithms in the literature [5,6] specifically address the camcorder capture attack, particularly in the context of digital cinema watermarking. In [5], Lubin et al. embed watermark patterns in low spatio-temporal frequencies. In [6], van Leest et al. modulate the mean luminance of the frames in time. Both methods are designed for base-band embedding and would require complexities similar to that of [3,4] when implemented for MPEG-2 bit-streams.

In this paper, we propose a new watermarking method for forensic tracking applications. The proposed method modifies only the quantization matrices, which are present in the MPEG-2 bit-stream[1]. As a result, it can be implemented with minimal computational complexity. Nonetheless, the method is robust to re-compression at lower bit-rates and camcorder capture attacks. The embedding and detection procedures are explained in Sec. 2. Implementation details and experimental results are presented in Sec. 3.

2. PROPOSED METHOD

2.1. Watermark Embedding

In the MPEG-2 and MPEG-4 (Part 2) compression standards, main encoding steps include: performing a DCT transform on 8×8 pixel (or prediction-error) blocks $\mathbf{X}$; and quantizing resulting DCT coefficients $\mathbf{C}$ according to a quantization matrix $\mathbf{Q}$. The quantization matrix indicates one quantization level for each frequency coefficient. Therefore, it allows for specifying higher step sizes (coarser quantization) at higher frequencies where the human visual system is less sensitive. We have

$$\begin{aligned} \mathbf{C} &= \mathrm{DCT}(\mathbf{X}) && (1) \\ \mathbf{C}_q &= \mathrm{round}\left(\mathbf{C}./\mathbf{Q}\right), && (2) \end{aligned}$$

where $\mathbf{C}_q$ denote quantized DCT coefficients and ./ is element-wise division. The quantization matrices are used during decoding to scale quantized DCT coefficients back to their reconstructed values $\mathbf{C}_r$, which in turn are transformed to reconstructed pixel (or prediction error) blocks $\mathbf{X}_r$:

$$\begin{aligned} \mathbf{C}_r &= \mathbf{C}_q. * \mathbf{Q} && (3) \\ \mathbf{X}_r &= \mathrm{IDCT}(\mathbf{C}_r), && (4) \end{aligned}$$

where $.*$ represents element-wise multiplication. Based on perceptual experiments and video signal statistics, a default quantization matrix is selected for each standard. However, both MPEG-2 and MPEG-4 (Part 2) standards allow the encoder to specify a custom matrix in the bit-stream. In general these matrices are specified in the `sequence_header` field of the bit-stream along with other sequence specific data such as picture size. Standards also allow different custom matrices to be transmitted for different sections of the content by repeating the `sequence_header` such that changes in the content characteristics can be accommodated. A quantization matrix specified in one `sequence_header` remains valid (used to decode all consecutive frames) until the next `sequence_header`. Note that current DVB and DVD standards require the presence of a `sequence_header` every 0.6 seconds or less. This requirement enables decoders to start decoding the bit-stream with no more than a 0.6 second delay. (For instance, when the channel is changed or the user seeks an arbitrary point in the movie.) Often a `sequence_header` is inserted for each GOP (12-15 pictures).

Our watermark embedding process is based on modulating these quantization matrices in time according to a watermarking pattern $\mathbf{w}$ that carries payload information, such as a user ID. In particular, we identify the `sequence_header` in the bit-stream by searching for the `sequence_start_code`. We read the custom matrix from the bit-stream $\mathbf{Q}$, modify some elements of this matrix and write the modified matrix $\mathbf{Q}'$ back into the bit-stream[2].

The watermark pattern is obtained by pseudo-randomly generating M base-sequences $\mathbf{b}_m$ of length L_w over the alphabet $\{-1, +1\}$, circularly shifting each according to a part of the payload information pL_i, i.e. $\mathbf{w}_m = \mathrm{CircShift}(\mathbf{b}_m, pL_m)$, and concatenating circularly shifted sequences: $\mathbf{w} = [\mathbf{w}_0 \mathbf{w}_1 \cdots \mathbf{w}_M]$.

The modification step groups quantization matrix elements into two parts based on the direction of corresponding frequency components. The horizontal frequencies (top-right triangle of the matrix) and vertical frequencies (bottom-left triangle) are modified with opposite polarities:

$$Q'(u,v) = \begin{cases} Q(u,v)(1+\alpha w[n]) & \text{if } u < v \\ Q(u,v)(1-\alpha w[n]) & \text{if } u > v \\ Q(u,v) & \text{otherwise,} \end{cases} \qquad (5)$$

where $u, v \in \{0, 1, \cdots, 7\}$ are horizontal and vertical frequency indexes, $w[n] \in \{-1, +1\}$ is the watermark symbol for the n^{th} sequence header (GOP) and α is the watermark strength.

2.2. Watermark Detection

During watermark embedding, we have modified selected elements of the quantization matrix for each GOP. When a quantization level is modified in the bit-stream, during decoding the corresponding coefficient is de-quantized with a different than intended value. If the quantization level is increased (decreased), then the magnitude of the reconstructed coefficient also increases (resp. decreases):

$$\begin{aligned} C_r'(u,v) &= C_q(u,v)Q'(u,v) && (6) \\ &= C_q(u,v)Q(u,v)(1 \pm \alpha w[n]) && (7) \\ &= C_r(u,v)(1 \pm \alpha w[n]). && (8) \end{aligned}$$

[1] The technique is also applicable to other compression methods that utilize similar quantization matrices.

[2] The MPEG syntax allows the use of a default matrix (stated in the standard) by resetting a flag without the need for explicitly placing it in the bit-stream. In this case, we modify the default matrix and set the flag before inserting the modified matrix into the bit-stream. This process will slightly increase the size of the bit-stream.

Therefore, the impact of changing quantization matrices will be apparent in the magnitude/energy of the reconstructed DCT coefficients.

The watermark detection is performed in the base-band using side-information about the original bit-stream as follows:
i) The received video and the original are temporally aligned, i.e. for each received frame, we determine the corresponding frame in the original video. This process can be implemented very efficiently by using video fingerprinting (robust hashes).
ii) We divide the video into segments $\mathbf{S}[n]$ corresponding to the GOPs used in the original bit-stream. This requires side-information (a list of GOP boundaries) about the original bit-stream. We assume that the side-information is kept along with the fingerprints in a database.
iii) The mean energy of DCT blocks is computed for each segment. This requires computing the 8×8 block DCT for all frames (and all blocks within a frame) and accumulating the square of each coefficient

$$\overline{\mathbf{E}[n]} = \sum_{\mathbf{X} \in \mathbf{S}[n]} \mathrm{DCT}(\mathbf{X})^2 \ . \tag{9}$$

iv) We estimate the watermark symbol of the current GOP $w[n]$ from energies of horizontal and vertical frequency contents. Recall that (Eqn. 5) the predominantly horizontal ($u < v$) and predominantly vertical ($u > v$) frequency components were modified in opposing directions (by $\pm\alpha w[n]$). The estimate is obtained by

$$\widetilde{w}(u,v) = \frac{\overline{E(u,v)} - \overline{E(v,u)}}{\overline{E(u,v)} + \overline{E(v,u)}} \tag{10}$$

for any $(u,v) : u < v$.

Assuming the received video is watermarked, i.e. $\overline{E(u,v)} = \overline{C_r'(u,v)^2}$ and horizontal and vertical frequency components are of similar strength, i.e. $\overline{C_r(u,v)^2} \approx \overline{C_r(v,u)^2} \approx \overline{C_r^2}$, we can re-write the estimate as

$$\begin{aligned}
\widetilde{w}(u,v) &= \frac{\overline{C_r'(u,v)^2} - \overline{C_r'(v,u)^2}}{\overline{C_r'(u,v)^2} + \overline{C_r'(v,u)^2}} && (11)\\
&= \frac{\overline{[C_r(u,v)(1+\alpha w)]^2} - \overline{[C_r(v,u)(1-\alpha w)]^2}}{\overline{[C_r(u,v)(1+\alpha w)]^2} + \overline{[C_r(v,u)(1-\alpha w)]^2}} && (12)\\
&\simeq \frac{\overline{C_r^2}[(1+\alpha w)^2 - (1-\alpha w)^2]}{\overline{C_r^2}[(1+\alpha w)^2 + (1-\alpha w)^2]} && (13)\\
&\simeq 2\alpha w. && (14)
\end{aligned}$$

Note that the latter assumption holds if enough pixel blocks are averaged as there is no apparent bias toward horizontal or vertical components. The experimental results in the next section verify the validity of this assumption.

When all watermark symbol estimates $\widetilde{w}(u,v) : u < v$ are collected, we split them into vectors $\widetilde{\mathbf{w}}_m$ and correlate each with all possible circular shifts of the corresponding base-sequence. (This operation can be implemented efficiently using FFTs.) We select the watermark estimate for the frequency (u,v) that yields the best correlation. The location of the correlation peak indicates the embedded payload pL_m, whereas its normalized magnitude is a measure of the confidence or the false positive probability.

3. EXPERIMENTAL RESULTS

We implemented our system using $M = 4$ base-sequences of length $L_w = 512$, with each coding 8 bits of a 32 bit payload. The watermark embedding strength α is set to $1/8$. One repetition of the watermark sequence (length 2048) requires a content segment of approximately 20 minutes. We selected 7 MPEG-2 files of various lengths and bit-rates for testing (Table. 1).

Identifier	Type	Duration (hh:mm:ss)	Avg. Bit-rate
Seq. 1	NTSC	1:47:54	3.5 Mbps
Seq. 2	NTSC	2:27:14	5.8 Mbps
Seq. 3	PAL	2:28:34	4.7 Mbps
Seq. 4	PAL	2:55:35	4.6 Mbps
Seq. 5	PAL	1:19:05	5.4 Mbps
Seq. 6	PAL	1:33:39	5.0 Mbps
Seq. 7	PAL	7:59:12	2.0 Mbps

Table 1. Test set.

3.1. Perceptual Quality

Visual inspection of watermarked sequences both on computer monitors and on large-screen TV sets did not yield any perceptible artifacts. A sample frame is shown in Fig. 1.

Fig. 1. A sample watermarked frame.

We further computed the peak-signal-to-noise-ratio (PSNR) for each frame in the luminance channel, using the original MPEG-2 bit-stream as the reference. The mean and minimum PSNR values for each sequence is presented in Table. 2. Note that the PSNR values are sufficiently high and further support our visual inspection.

3.2. Robustness

We considered the following as two possible attack scenarios: *Perfect Capture* and *Camcorder Capture*.
Perfect Capture: Here, we assume the decoded bit-stream can be captured from some digital output port without any distortions. (We simulate this scenario by decompressing into a file.) Using widely available tools, each captured sequence is processed such that any

Identifier	Mean PSNR	Min. PSNR
Seq. 1	47.12	37.61
Seq. 2	49.92	39.53
Seq. 3	49.64	39.90
Seq. 4	48.61	38.98
Seq. 5	50.06	37.00
Seq. 6	50.82	36.57
Seq. 7	46.28	37.81

Table 2. Mean and minimum per frame PSNR (dB) on luminance channel.

black-bars or boundary pixels are removed, its width is reduced to 320 pixels, and it is compressed down to 300 kbps using the XVID codec. Detection is performed on these sequences without undoing spatial manipulations. Results are plotted in Fig. 2 for a detection period of 40 minutes. We see the confidence level (normalized correlation peak) and corresponding false positive probability on the left and right axes, respectively. In all cases, the watermark payload is correctly identified with very high confidence.

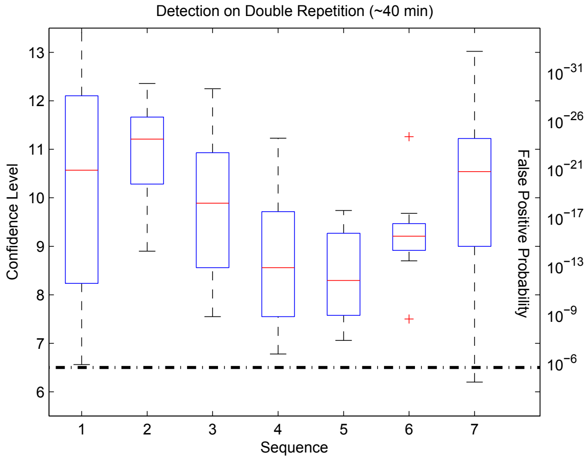

Fig. 2. Detection results over 40 min. $P_{fp} = 10^{-6}$ is set as the detection threshold.

Camcorder Capture: One of the watermarked sequences ($Seq.4$) is displayed on a flat-screen TV set and captured using a DV-camera. It is further transcoded into MPEG-2 at 6 Mbps. As seen in the sample frame in Fig. 3, the capture introduced arbitrary scaling, rotation, some perspective projection and blurring. Detection is performed after manually cropping the central portion of the frame containing the content and aligning it temporally. No action is taken to correct for spatial distortions. The watermark is detected and the correct payload is decoded with a confidence level of 7.87 corresponding to a false positive probability of $P_{fp} = 5.1 \cdot 10^{-11}$ from a 40 min clip.

4. CONCLUSION

We have proposed a robust MPEG-2 bit-stream watermarking algorithm which can be used to deter capture and unauthorized redistribution of electronically distributed movies. The algorithm has a very low complexity and is suitable for implementation even in current set-top boxes. In our threat model, we assumed that the bit-stream can be securely decoded and only considered capture attacks thereafter. An attacker with access to the watermarked bit-stream may potentially compromise the security of the watermark by inspecting the sequence headers. Once the sequence is decoded, however, it is significantly harder to estimate and remove the watermark. Under this threat model, we have shown that the watermark survives even after low bit-rate compression or capture with a video camera. Proposed detection period of 20-40 min is longer than, for instance, the digital cinema specification [7]. Nonetheless, multiple detection windows will be available for a typical movie. Currently, we are investigating possible extensions of the algorithm to new generation codecs such as H.264.

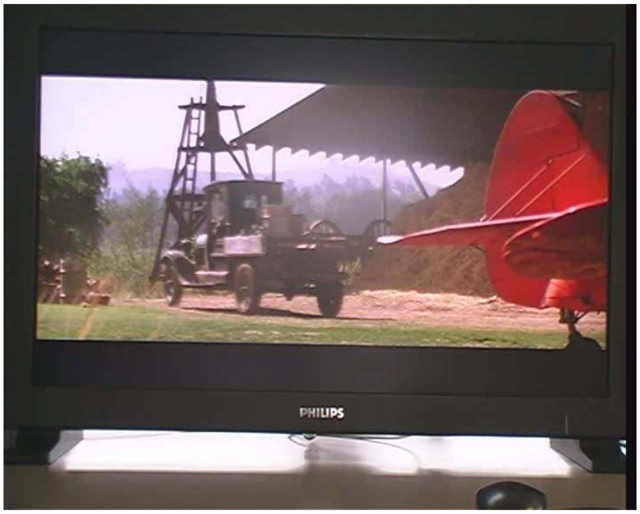

Fig. 3. Sample frame captured using a video camera.

5. REFERENCES

[1] ISO/IEC 13818-2, “Information technology – Generic coding of moving pictures and associated audio information: Video,” International Standard, 2000.

[2] ISO/IEC 14496-2, “Information technology – Coding of audio-visual objects – Part 2: Visual,” International Standard, 2004.

[3] F. Hartung and B. Girod, “Watermarking of uncompressed and compressed video,” *IEEE Sig. Proc. Mag.*, vol. 66, no. 3, pp. 283–301, May 1998.

[4] G. Langelaar and R. Lagendijk, “Optimal differential energy watermarking of DCT encoded images and video,” *IEEE Trans. Image Proc.*, vol. 10, no. 1, pp. 148–158, 2001.

[5] J. Lubin, J. A. Bloom, and H. Cheng, “Robust, content-dependent, high-fidelity watermark for tracking in digital cinema,” in *Proc. SPIE: Sec. Steg. and Watermarking of Mult. Cont. V*, E. J. Delp and P. W. Wong, Eds., vol. 5020, Jan. 2003.

[6] A. van Leest, J. Haitsma, and T. Kalker, “On digital cinema and watermarking,” in *Proc. SPIE: Sec. Steg. and Watermarking of Mult. Cont. V*, E. J. Delp and P. W. Wong, Eds., vol. 5020, Jan. 2003, pp. 526–535.

[7] Digital Cinema Initiatives, LLC, *Digital Cinema System Specification V1.0*, http://www.dcimovies.com/DCI-Digital-Cinema-System-Spec-v1.pdf, Std., 2006.