International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:10, 2007

Implementing an Adaptive Behavior for Spread Spectrum Watermarking Procedures

Franco Frattolillo

Abstract**—** The advances in multimedia and networking technologies have created opportunities for Internet pirates, who can easily copy multimedia contents and illegally distribute them on the Internet, thus violating the legal rights of content owners. This paper describes how a simple and well-known watermarking procedure based on a spread spectrum method and a watermark recovery by correlation can be improved to effectively and adaptively protect MPEG-2 videos distributed on the Internet. In fact, the procedure, in its simplest form, is vulnerable to a variety of attacks. However, its security and robustness have been increased, and its behavior has been made adaptive with respect to the video terminals used to open the videos and the network transactions carried out to deliver them to buyers. In fact, such an adaptive behavior enables the proposed procedure to efficiently embed watermarks, and this characteristic makes the procedure well suited to be exploited in web contexts, where watermarks usually generated from fingerprinting codes have to be inserted into the distributed videos "on the fly", i.e. during the purchase web transactions.

Keywords**—** Copyright protection, digital watermarking, intellectual property protection.

I. Introduction

Digital watermarking [1] is nowadays considered a main technology able to give an effective support to the copyright protection of MPEG videos distributed on the Internet [2]. To this end, many watermarking procedures exploit insertion schemes based on "fingerprinting techniques", which enable the copyright owner to insert a distinct watermark able to identify the buyer within any copy of video that is distributed [1], [3]. Thus, it should be possible to establish if a user is illegally in possession of a video as well as who has initially bought and then illegally shared it via, for example, peer-to-peer network applications [2]. Furthermore, in order to increase security and robustness, such watermarking procedures adopt "readable" watermarking schemes based on "blind" and not publicly available decoders [2], let the embedded watermark depend on the host signal, and adopt "anticollusion" codes, in which case the watermarked information and the embedding strategy are chosen in such a way that averaging different watermark signals, each identifying a different colluding user, leaves certain parts of the watermark unaffected, thus permitting the recovery of some information about the colluding user pool [3], [4].

However, watermarking procedures exploiting fingerprinting techniques are characterized by an "on buyer" behavior. In fact, since watermarks are tied to buyers, they have to be embedded into the videos to be protected upon the buyers' purchase requests. This means that, when purchase requests are sent to web content providers (CPs), the watermarks have to be embedded into the required videos "on the fly", i.e. during the purchase web transactions. This requires an efficient implementation of the watermarking procedures which does not compromise security and robustness. Furthermore, since a robust and secure watermarking procedure can be computationally intensive or increase the size of the protected videos, it is important to adapt it to the specific characteristics of both the terminals used to open the required videos and the transactions carried out between users and CPs. For example, a PDA or a mobile phone or a terminal with no storing capacity or limited visualization capacities could receive low-quality, "lightly watermarked" videos during transactions taking place on low performance networks. Finally, the effectiveness of anticollusion codes strictly depends on the length of the adopted codes [4], and this means that watermarking procedures have to enable the insertion of long fingerprinting codes without impairing the final quality of the watermarked videos. Therefore, an advanced, web oriented watermarking procedure for MPEG videos should:

- provide a good degree of robustness against the most common, nonmalevolent manipulations;
- survive intentional attacks;
- implement a readable scheme based on a blind and not publicly available decoder;
- depend on the host signal and exploit "anticollusion" codes;
- be characterized by an efficient implementation that does not compromise security and robustness;
- directly operate on the compressed bit-stream, so as not to limit the performance of the implementation;
- exhibit an "adaptive" behavior, that is, it should take into account the characteristics of both the terminals used to open the videos and the network transactions carried out to deliver them to the respective buyers.

Although many techniques for embedding fingerprints in MPEG videos have been proposed in literature, spread-spectrum additive embedding techniques (SS) have proven robust and secure against a number of signal processing operations and attacks [5], [6], [7]. Moreover, with appropriately chosen parameters and adopting specific improvements [8], the spread-spectrum watermark can survive moderate geometric distortions without suffering from the sensitivity to amplitude scaling evidenced by other well-known watermarking

F. Frattolillo is with the Research Centre on Software Technology, Department of Engineering, University of Sannio, Benevento, Italy (phone: +39 0824 305806; fax: +39 0824 305840; e-mail: frattolillo@unisannio.it).

procedures, such as those based on the "quantization index modulation" (QIM) [9], and roughly achieving the same noise robustness gain as QIM. Furthermore, since an SS embedding technique usually depends on a few parameters, it can be easily exploited to implement an adaptive behavior able to match the requirements reported above.

This paper describes how a simple and well-known watermarking procedure based on a spread spectrum method and a watermark recovery by correlation [10], [11], [12] can be improved to effectively protect MPEG-2 videos distributed on the Internet. In fact, the procedure directly acts on compressed video streams and has been provided with an adaptive, on buyer insertion scheme matching the requirements reported above.

The paper is organized as follows. Section II describes the proposed watermarking procedure. Section III reports on some experimental results. Finally, Section IV reports conclusion remarks.

II. The watermarking procedure

The proposed watermarking procedure is based on the approach described in [8], and is specialized for MPEG-2 compressed video streams. In particular, the embedding approach is an improvement of the original SS techniques described in [10], [11], [12], and is based on the key idea of removing the host signal as source of interference, thus producing a dramatic improvement in the quality of the watermarking process.

A. The basic scheme

In the basic scheme shown in Figure 1, the insertion of the watermark in the compressed video V is accomplished by extracting the encoded 8×8 blocks from the video and processing them together with the corresponding blocks of the watermarking signal W. In particular, the MPEG-2 bitstream is split into its main components, and only the DCT encoded signal blocks are modified.

Each encoded block is represented by a sequence of Huffman codes, each representing one (run-level)-pair and, thus, one quantized non-zero DCT coefficient of the current signal block. Therefore, to insert the watermark, each Huffman code is decoded (EC^{-1}) and inversely quantized (Q^{-1}), i.e. the mapping from the quantizer index to the quantizer representative is performed. After this processing, a quantized DCT is added to the corresponding DCT coefficient from the transformed W signal, yielding a watermarked DCT coefficient. This is then quantized (Q) and Huffman encoded (EC).

It is worth noting that the basic watermarking scheme has been designed not to increase the output bit-rate. Therefore, in Figure 1, the output 2 is selected only if the number of bits used to represent the codeword for the protected video signal is less or equal than the number of bits used to represent the same codeword in the original video signal [11].

Finally, the watermarking procedure is also completed by a scheme for drift compensation, which is not shown in Figure 1 for the sake of brevity.

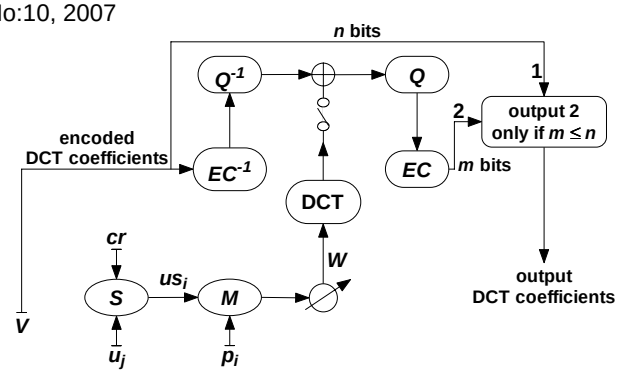

Fig. 1 The basic watermarking scheme

B. The "on buyer" behavior

The proposed procedure adds a noise-like signal to the encoded video signal processed block by block. As shown in Figure 1, the watermark signal is generated from a sequence of bits $u_j \in \{-1, 1\}$, which is used to identify a user and is spread (S) by a large factor cr, called *chip-rate*, thus obtaining the spread sequence $us_i = u_j$, with $j \cdot cr \leq i < (j+1) \cdot cr$. Then, the noise-like signal is generated by modulating (M) the spread sequence with a binary pseudo-noise sequence $p_i \in \{-1, 1\}$, which, in the proposed solution, has to be unambiguously associated to the protected video. Thus, once a protected video has been selected, it is possible to employ the pseudo-noise sequence p_i associated to it to extract the watermark and thus obtain the user sequence $u_j \in \{-1, 1\}$. To this end, the signal of the protected video can be correlated with the p_i sequence over a cr wide correlation window, and the extracted watermark can be then analysed to obtain the sequence of bits identifying the user who bought the video.

Finally, it is worth noting that the sequences u_j are assigned to identify users according to an anticollusion technique [4], [13] and exploit an error correction code. However, this issue is not elaborated here because this is not a main goal of the paper and for the sake of brevity.

C. The improved scheme

The scheme described in Sections II-A and II-B is based on the simple formula

$$\mathbf{s} = \mathbf{x} + u\mathbf{m} \tag{1}$$

where the vector $\mathbf{x}$ is the host signal, $\mathbf{m}$ is the chip sequence built from p_i, u represents a bit from the u_j sequence, and the vector $\mathbf{s}$ is the watermarked signal. In particular, (1) assumes that one bit of information from the u_j sequence is embedded in the vector $\mathbf{s}$ of cr values according to the common SS techniques. However, the implemented watermarking scheme is based on a slight modification to the SS approach, which is defined in [8] as the "linear" version of the improved SS technique (ISS). In fact, this variant assumes that the amplitude of the inserted chip sequence can vary by a linear function

$$\mathbf{s} = \mathbf{x} + (\alpha u - \lambda x)\mathbf{m} \tag{2}$$

where $x \triangleq \langle \mathbf{x}, \mathbf{m} \rangle / \langle \mathbf{m}, \mathbf{m} \rangle$ and $\langle \mathbf{x}, \mathbf{m} \rangle$ is the inner product defined as

$$\langle \mathbf{x}, \mathbf{m} \rangle \triangleq \frac{1}{cr} \sum_{i=0}^{cr-1} x_i m_i \tag{3}$$

In particular, (3) also defines the norm whenever it is used, for example, as $\langle \mathbf{x}, \mathbf{x} \rangle$.

The parameters α and λ control the distortion level and the removal of the carrier distortion on the detection statistic. In fact, if $\mathbf{y}$ is the available distorted version of $\mathbf{s}$ obtained by adding to $\mathbf{s}$ a noise $\mathbf{n}$ modelled as an uncorrelated white Gaussian random process, the sufficient statistic available at the watermark extractor r is

$$r = \frac{\langle \mathbf{y}, \mathbf{m} \rangle}{\langle \mathbf{m}, \mathbf{m} \rangle} = \alpha u + (1 - \lambda x) + n \tag{4}$$

where $n \triangleq \langle \mathbf{n}, \mathbf{m} \rangle / \langle \mathbf{m}, \mathbf{m} \rangle$. Therefore, by using the encoder knowledge about the signal, the performance of the watermarking system can be enhanced by modulating the energy of the inserted watermark to compensate for the host signal interference. In particular, the closer λ is made to 1, the more the influence of x is removed from r.

In addition, the detector is the same as in the SS watermarking techniques, i.e., the detected bit is $sign(r)$. Furthermore, traditional SS techniques can be obtained by setting $\alpha = 1$ and $\lambda = 0$.

The results reported in [8] make it possible to calculate the optimal values of α and λ for the watermarking system defined by (2) and under the assumptions made in Sections II-A and II-B. In particular, low values for the error probability (i.e. lower than 10^{-5}) can be achieved by setting

$$\alpha = \sqrt{\frac{cr - \lambda^2 \sigma_x^2}{cr}} \tag{5}$$

and λ close to 1 (i.e. in the range $0.9, 1$) under the assumption that cr is large enough and SNR is higher than 10 db (decibels).

D. The adaptive behavior

As reported in Section I, watermarking procedures should be characterized by an adaptive behavior. To this end, in the proposed procedure, the watermark embedded in a video depends on the characteristics of both the terminal used to open the video and the quality of the network connection established between the user and the CP. This dependence, as shown in Figure 2, is controlled by two specific functions, Φ and Ψ, which determine the chip-rate cr and the video output bit-rate respectively. These functions depend on two variables, τ and η, which qualify the user terminal type and network connection respectively.

In the proposed model, τ essentially captures the terminal visualization capacities, i.e. the video resolution, whereas η synthesizes the bandwidth and latency of the user network connections. In fact, τ can be derived from what declared by users when they interact with the web servers of CPs, whereas η can be also directly estimated by CPs during the transaction

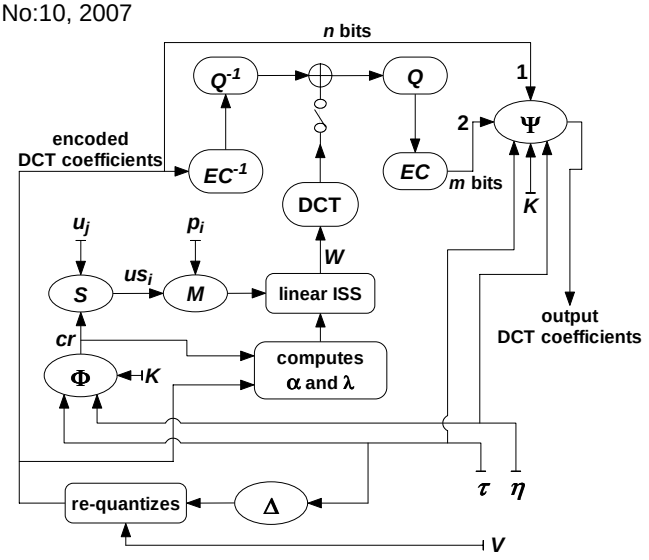

Fig. 2 The improved watermarking scheme

web phase with users. In particular, network connections are roughly differentiated in three main categories: modem and GPRS links, DSL and LAN connections, and T1, T3 lines.

The conducted tests have shown that cr can usefully vary in the range from $cr_{min} = 10,000$ to $cr_{max} = 1,000,000$. Therefore, by setting $\tau_{min} = 320 \times 240$ and $\tau_{max} = 1024 \times 768$, Φ can be expressed as follows:

$$\Phi = \begin{cases} cr_{min} & \text{if } \tau < \tau_{min} \\ cr_{max} & \text{if } \tau > \tau_{max} \\ \eta\left(\frac{(cr_{max} - cr_{min})(\tau - \tau_{min})}{\tau_{max} - \tau_{min}} + K cr_{min} \right) & \text{otherwise} \end{cases} \tag{6}$$

In (6) η and K may respectively assume only three different values, each corresponding to a different kind of user network connection. Table I reports the possible values for η and K derived from the conducted tests. In fact, the product $\eta \cdot K$ can be assumed as a relative weight able to characterize the network connections.

TABLE I
THE POSSIBLE VALUES OF η AND K IN (6)

	η	K
modem or GPRS links	0,2	5
DSL, LAN connections	0,7	10
T1, T3 lines	0,5	100

As reported above and in Figure 2, Ψ controls the video output bit-rate. It specifies the maximum increment percentage that the output bit-rate can induce in the video size. The conducted tests have shown that such increment can usefully vary in the range from $in_{min} = 5\%$ to $in_{max} = 50\%$ without compromising the final video quality. Therefore, Ψ can be

specified as follows:

$$\Psi = \begin{cases} in_{min} & \text{if } \tau < \tau_{min} \\ in_{max} & \text{if } \tau > \tau_{max} \\ \eta \left(\frac{(in_{max} - in_{min})(\tau - \tau_{min})}{\tau_{max} - \tau_{min}} + K in_{min} \right) & \text{otherwise} \end{cases} \quad (7)$$

As in (6), also in (7) η and K may respectively assume only three different values, each corresponding to a different kind of user network connection. Table II reports these values. Moreover, the product $\eta \cdot K$ can be still assumed as a relative weight able to characterize the network connections. However, the weights in Table II are less than the corresponding ones reported in Table I, and this because the possible range for the chip-rate is larger than the range specifying the increment percentage of the video size.

TABLE II
THE POSSIBLE VALUES OF η AND K IN (7)

	η	K
modem or GPRS links	0,2	5
DSL, LAN connections	0,4	8
T1, T3 lines	0,3	24,3

Φ and Ψ have been determined taking into account that a high value for cr increases the watermark robustness, but at the same time decreases the data rate for watermark. On the other hand, controlling the bit-rate means determining the fraction of the watermark signal that can be successfully embedded in the videos to be protected: increasing the bit-rate means to increase this fraction and thus improve the robustness of the watermarking, even though the video quality could suffer a degradation. To this end, it is worth noting that, in many well-known procedures, the watermarking is assumed not to increase the output bit-rate [11], [12]. On the contrary, in the proposed procedure, Ψ may increase the bit-rate, since the video size can change according to both the required protection level and the actual service conditions: the former is essentially identified by τ, whereas the latter are captured by η. Therefore, once the increment for a video size has been determined, Ψ sets a counter to the increment value. Then, Ψ updates the counter by subtracting from it the difference between the number of bits needed to represent a codeword for the watermarked signal sent to output and the number of bits used to represent the same codeword for the original video signal: positive differences are considered "debts", whereas negative differences are considered "credits". Therefore, when the counter reaches 0, further codewords for the watermarked signal are sent to output only if further credits occur that balance debts. Thus, the procedure ensures that the increment of the video size remains constant.

In the proposed procedure, cr may vary to implement the adaptive behavior. Therefore, to extract the watermark from a video, it is necessary to specify the associated sequence p_i and the value of cr used to watermark the video. To this end, watermarking is actually performed in two phases. In the former, a cr_v value constantly associated to the video is used to embed the first n values of the sequence u_i. These values are used to identify the chip-rate cr calculated by Φ and that has to be used to watermark, in the latter phase, the remaining part of the video. Thus, given the video, the sequence p_i and the value cr_v can be identified and then applied to retrieve the first n values of the sequence u_j, which identify the cr value to be used to extract the watermark from the remaining part of the video.

The adaptive behavior of the proposed procedure is further improved by assuming that the distributed videos can be characterized by a different quality depending on the visualization capacities of user terminals. This feature is implemented by stating that the original video quality directly depends on τ. To this end, it is worth noting that the adaptive, on buyer behavior requires that a content manipulation is performed "on the fly", when the purchase web transaction takes place, in order to adapt the video quality and the applied protection to the transaction characteristics. In particular, in order not to reduce the robustness and security levels achievable by the watermarking procedure, watermark has to be embedded after the video quality adaptation, and this means that different versions of the available videos should be handled at the CP side. In fact, two main solutions can be adopted by CPs: the former requires that different versions of each video made available by a CP are generated, stored and handled at server side, whereas the latter is based on the dynamic generation of such versions from high quality master videos. However, holding one version of a video for each possible quality level is a very heavy solution at server side, particularly when the CP server has to address low to high resolution video terminals. On the contrary, the latter solution appears to be more flexible and memory saving, provided that an efficient implementation of the adaptation procedure is used.

Quality adaptation of MPEG-2 videos can be carried out by exploiting one of the two main and well-known techniques: the re-quantization of the DCT coefficients and the cut of the high frequencies, i.e. the AC coefficients [14]. The former is based on the increment of the quantization step in order to pull down ulterior DCT coefficients, whereas the latter is simply based on eliminating ulterior terms of every DCT 8×8 blocks by cutting the terms relative to the high frequencies. Therefore, both techniques reduce the dimensions of the bit-stream as well as the quality of the video, even if it is demonstrated that the former technique turns out to be more efficient than the latter in that it produces a smaller quantization error.

MPEG-2 video re-quantization is therefore controlled by the function Δ, which determines the increment of the re-quantization step. The conducted tests have shown that such increment may usefully vary in the range from 0 to $ir_{max} = 30\%$. Therefore, Δ can be calculated as:

$$\Delta = \begin{cases} ir_{max} & \text{if } \tau \leq 320 \times 240 \\ 0 & \text{if } \tau > 640 \times 480 \\ ir_{max} \left(1 - \left(\frac{\tau - (320 \times 240)}{(640 \times 480) - (320 \times 240)} \right)^2 \right) & \text{otherwise} \end{cases} \quad (8)$$

Obviously, a re-quantization step equals to 0 means that the original master video quality has not to be modified. Furthermore, the non linear behavior of (8) allows for mostly reducing the quality of the low resolution videos, i.e., the videos that have to be lightly watermarked.

Finally, it is worth noting that re-quantization results in being strategic in order to implement the adaptive behavior of the proposed procedure. In fact, whenever a malicious user attempts to obtain a lightly watermarked video by deceptively claiming to be provided with a low resolution video terminal and to be connected by means of a low performance link, he/she ends up obtaining only a re-quantized, low quality video which, even if unprotected, can be neither advantageously played by a high resolution video terminal nor considered interesting to Internet "pirates".

III. Experimental results

The proposed watermarking procedure has been assessed by performing some relevant attacks that attempt to render the embedded watermark not readable. In particular, Tables III, IV and V summarize the results obtained respectively under three different attacks: the IBM attack, frame dropping and frame averaging.

The first attack is considered an "ambiguity attack" in that it attempts to discredit the authority of the watermark by embedding one or several additional watermarks such that it is unclear which was the first, authoritative watermark [15]. However, in this context, the IBM attack is exploited to add noise to videos so as to obscure the original watermarks. The second attack can be considered a "simple attack" or a "detection-disabling attack" in that it attempts to impair the embedded watermark by manipulations of the whole watermarked data (host data plus watermark), without an attempt to identify and isolate the watermark [12]. The third attack is a "removal attack" in that it attempts to estimate the watermark, separate the watermarked data into host data and watermark, and discard only the watermark [12].

More than 80 videos have been used for the conducted tests. In particular, the tested videos are coded in MPEG-2, at 30 fps, with an original resolution of 1024×768 pixels. Their duration is in the range between 60 and 120 seconds.

For each attack, two main values have been calculated: the bit error rate (ber) affecting the watermark extraction, and the peak signal-to-noise ratio ($\overline{psnr}$). In particular, the $\overline{psnr}$ has been estimated as the mean of the $psnr$ values calculated over all the I-frames contained in the watermarked and the attacked video. To this end, each $psnr$ value has been calculated by using the following definition: $10\log\left(255^2/MSE\right)$, where MSE is for the "mean squared error" computed on an I-frame belonging to the watermarked and attacked video. Therefore, the $\overline{psnr}$ can estimate the quality of the two compared videos.

In the following tables, two values are reported under different values of the video terminal resolution (τ) and of the network connection (η). The first is the BER value, defined as the mean of the ber values calculated over the tested videos and expressed in percentage. The second is the $PSNR$ value, defined as the mean of the $\overline{psnr}$ values calculated over the tested videos and expressed in decibels.

TABLE III
The results of the IBM attack

	320×240	640×480	1024×768
modem or GPRS link	22.1% 30,2db	15.3% 33,5db	9.4% 32,7db
DSL, LAN	19.3% 28,7db	11.5% 32,1db	6.1% 31,5db
T1, T3	15.9% 27,3db	6.5% 30,9db	3.6% 29,6db

The user sequence u_j employed in all the conducted tests is 64 bit long, even though only 32 bits have to be considered actually used to identify a user by means of an anticollusion code. In fact, the remaining 32 bits are exploited as "check bits" needed to implement an error-correcting code. Therefore, if the ber value results in being less or equal to 9% (6 bit error) in extracting the watermark from a video, the user sequence inserted in the video can be correctly re-built. Furthermore, it is worth noting that a $\overline{psnr}$ value equal to 35 decibels is nowadays widely assumed as a lower limit for the video quality in a commercial scenario, according to the current literature [16]. Therefore, an attack can be considered valid only if the obtained ber value is greater than 9% and the $\overline{psnr}$ is greater than 35 db. Obviously, the limit of 35 db has not to be considered a hard video quality threshold, but only an estimate.

Table III shows that the proposed procedure achieves a good performance under the IBM attack. In fact, for low values of τ and η, the BER is high, but the final video quality results low because the video, due to the re-quantization, is not able to contain the further information needed to make the watermark not readable. On the contrary, for high values of τ and η, the procedure results in being secure, and the attack cannot impair the embedded watermark: the BER values are prevalently less than 9%. It is also worth noting that the $PSNR$ values tend to assume lower values when τ and η become high, and this because the amount of information embedded, in this hypothesis, by the watermarking procedure and by the performed attack increases, thus exceeding the video capacity.

TABLE IV
The results of the frame dropping attack

	320×240	640×480	1024×768
modem or GPRS link	26.2% 33,4db	19.1% 39,3db	13.2% 37,5db
DSL, LAN	23.1% 32db	14.5% 38,4db	10.6% 35,3db
T1, T3	17.5% 30,8db	9.1% 37,7db	8.3% 33,2db

The frame dropping attack attempts to disable the watermark extraction by removing trunk of frames. In particular, when the dropping rate of video frame is high, errors are introduced to the whole watermark, making the performance of the procedure poor. However, this also leads to a significant damage to the video, and the results reported in Table IV, obtained under a value of frames dropped about 20%, reflect this condition. In particular, the successful attacks performed under some values of η and τ can be contrasted by increasing the number of the check bits used to implement the error-correcting code.

TABLE V
THE RESULTS OF THE STATISTICAL AVERAGING ATTACK

	320×240	640×480	1024×768
modem or GPRS link	13.4% 40,3db	12.5% 39,6db	9.2% 39,2db
DSL, LAN	10.5% 38,5db	9.6% 38,2db	7.2% 38,4db
T1, T3	9.3% 36,3db	5.6% 37,6db	3.1% 37,7db

In the statistical averaging attack, a high number of watermarked frames are collected so as the watermark can be estimated by statistical averaging. The attack has been performed by colluding about the 70% of the available frames, and the obtained results are shown in Table V. In particular, the procedure exhibits a good performance, and this is essentially due to its adaptive behavior, which can balance the final video quality with the achieved protection level.

Finally, it is worth noting that the $PSNR$ values obtained during the conducted tests demonstrate that the procedure can successfully protect the videos as well as reduce the final video quality to the allowed minimum values. In fact, one of the interesting aspect of the procedure, emerged from the test phase, is that, if the re-quantization phase reduces the video quality to a $\overline{psnr}$ value close to a predefined lower limit, such that of 35 db, and the subsequent watermark embedding is carried out taking care of saturating the video capacity without further reducing the final value of the $\overline{psnr}$, attacks to impair the embedded watermark end up obtaining $\overline{psnr}$ values much lower than the assumed limit, thus degrading the final video quality.

IV. Conclusions

This paper describes a watermarking procedure for the copyright protection of MPEG-2 videos distributed on the Internet. The procedure directly acts on compressed video streams and is implemented as an on buyer variant of the improved spread spectrum scheme described in [8]. The procedure can employ long anticollusion codes to increase security against average and collusion attacks, and is characterized by a novel adaptive behavior that is able to modulate the applied protection depending on both the terminals used to open the watermarked videos and the network transactions carried out to deliver them to buyers.

The experimental results confirm that a simple spread spectrum based watermarking procedure can be made robust and secure against a variety of manipulations by performing some improvements that do not penalize efficiency. This makes the procedure suitable to be exploited in web contexts, where an "on the fly" behavior is required. Moreover, the adaptive behavior of the procedure allows for achieving a trade-off between protection needs and the final quality of the distributed videos. Thus, whenever attacks attempt to impair the embedded watermarks, the final video quality ends up being degradated, thus making the attacked videos useless in commercial web applications and not interesting to Internet "pirates".

References

[1] I. Cox, J. Bloom, and M. Miller, *Digital Watermarking: Principles & Practice*. Morgan Kaufman, 2001.
[2] M. Barni and F. Bartolini, "Data hiding for fighting piracy," *IEEE Signal Processing Magazine*, vol. 21, no. 2, pp. 28–39, 2004.
[3] M. Wu *et al.*, "Collusion-resistant fingerprinting for multimedia," *IEEE Signal Processing Magazine*, vol. 21, no. 2, pp. 15–27, 2004.
[4] W. Trappe, M. Wu, *et al.*, "Anti-collusion fingerprinting for multimedia," *IEEE Trans. on Signal Processing*, vol. 41, no. 4, pp. 1069–1087, 2003.
[5] I. Cox, J. Kilian, *et al.*, "Secure spread spectrum watermarking for multimedia," *IEEE Trans. on Signal Processing*, vol. 6, no. 12, pp. 1673–1687, 1997.
[6] C.-Y. Lin, M. Wu, J. Bloom, I. Cox, M. Miller, and Y. Lui, "Rotation, scale and translation resilient watermarking for images," *IEEE Trans. on Image Processing*, vol. 10, no. 5, pp. 767–782, 2001.
[7] J. Lubin, J. Bloom, and H. Cheng, "Robust, content-dependent, high-fidelity watermark for tracking in digital cinema," in *Electronic Imaging 2003, Security and Watermarking of Multimedia Contents*, ser. SPIE Proceedings, P. W. Wong and E. J. Delp, Eds., vol. 5020, S. Jose, CA, USA, Jan. 2003, pp. 536–545.
[8] H. S. Malvar and D. A. F. Florêncio, "Improved spread spectrum: A new modulation technique for robust watermarking," *IEEE Trans. on Signal Processing*, vol. 51, no. 4, pp. 898–905, 2003.
[9] B. Chen and G. Wornell, "Quantization index modulation: a class of provably good methods for digital watermarking and information embedding," *IEEE Trans. on Inforormation Theory*, vol. 47, no. 4, pp. 1423–1443, 2001.
[10] F. Hartung and B. Girod, "Digital watermarking of raw and compressed video," in *Proc. European EOS/SPIE Symposium on Advanced Imaging and Network Technologies*, Berlin, Germany, October 1996.
[11] ——, "Digital watermarking of MPEG-2 coded video in the bitstream domain," in *Procs of the Int'l Conference on Acoustics, Speech, and Signal Processing*, vol. 4, Munich, Germany, 1997, pp. 2621–2624.
[12] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Procs of the IEEE*, vol. 87, no. 7, pp. 1079–1107, 1999.
[13] D. Boneh and J. Shaw, "Collusion-secure fingerprinting for digital data," *IEEE Trans. on Information Theory*, vol. 44, no. 9, pp. 1897–1905, 1998.
[14] Z. Lei and N. D. Georganas, "Rate adaptation transcoding for precoded video streams," in *Procs of the 10th ACM Int'l Conference on Multimedia*, Juan-les-Pins, France, 2002, pp. 127–136.
[15] F. Hartung, J. Su, and B. Girod, "Spread spectrum watermarking: Malicious attacks and counterattacks," in *Electronic Imaging 1999, Security and Watermarking of Multimedia Contents*, ser. SPIE Proceedings, vol. 3657, S. Jose, CA, USA, Jan. 1999, pp. 147–158.
[16] Y. Wang, J. Ostermann, and Y. Zhang, *Video Processing and Communications*. Prentice Hall, 2002.