# NEW ROBUST WATERMARKING SCHEME FOR VIDEO COPYRIGHT PROTECTION IN THE SPATIAL DOMAIN

Radu Ovidiu PREDA[1], Nicolae VIZIREANU[2]

*Această lucrare propune o schemă nouă de watermarking video pentru protecția drepturilor de autor. Watermark-urile folosite sunt imagini binare şi sunt inserate prin cuantizare în blocuri de luminanță. Fiecare bit al watermark-ului este împrăştiat într-un bloc de pixeli prin intermediul unei chei secrete. Imaginea şi watermark-ul original nu sunt necesare pentru decodare. Am testat rezistența algoritmului la 7 tipuri de atacuri diferite şi am îmbunătățit Rata Erorii de Bit la decodare prin folosirea codurilor corectoare de erori şi prin inserarea cu redundanță spațială şi temporală. Rezultatele experimentale demonstrează că watermark-ul inserat este invizibil şi rezistent la atacuri.*

*This paper proposes a new digital watermarking scheme for video copyright protection. The watermarks used are binary images and are embedded in blocks of luminance pixels by quantization. Every bit of the watermark is spread over a block of pixels with the use of a key. The original image and watermark are not needed for watermark extraction. We have tested the resilience of the algorithm against 7 different attacks and improved the decoding BER by using error correction codes and spatial and temporal redundancy. Experimental results show that the embedded watermark is invisible and robust to attacks.*

**Keywords:** digital video watermarking, copyright protection, spread spectrum watermarking, resilience to attacks

## 1. Introduction

The explosive growth of Internet and communication networks has led to the tremendous use of multimedia data like image, audio and video. Furthermore, due to the availability of tools to manipulate digital multimedia especially digital images, tampering of such data has become very easy. In this context, it is important to ensure the integrity and protection against unauthorized duplication of images and videos. A common technique is digital watermarking, a process by which a user-specified signal (watermark) is hidden or embedded into another signal (cover data), for example digital content such as electronic documents,

[1] Prof., Faculty of Electronics, Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania, e-mail: radu@comm.pub.ro
[2] Prof., Faculty of Electronics, Telecommunications and Information Technology, University POLITEHNICA of Bucharest, Romania, e-mail: nae@comm.pub.ro

images, sounds and video. There is urgent demand for techniques to protect the original digital data and to prevent unauthorized duplication or tampering.

The important requirements of such watermarks are imperceptibility, robustness and security. Watermark imperceptibility means that the watermark should be hidden in the cover image in such a way that it cannot be seen. Robustness of a watermark is the ability to extract the watermark correctly even if intentional or unintentional attacks are made on the watermarked image. To ensure security, only the authorized user should be allowed to embed and extract the watermark. This could be achieved by employing a cryptographic key while embedding.

The application that attracts the most attention is copyright protection. In this context, a watermark is permanently embedded in the work to identify its original owner. In order to be efficient, the embedded mark has to be *robust*, that is, it has to be detectable as long as the host carries its information, hence, the name of robust watermarking.

The most important issues in video watermarking are the invisibility of the watermark and the resilience of watermarking to attacks. A lot of watermarking methods have been proposed to improve both of these aspects. Extensive surveys dealing with video watermarking applications, attacks and methodologies can be found in [1].

Procedures that can recover the hidden mark without the use of the original unmarked data are defined as *blind* decoding. Some of these blind techniques require access to a reference key to extract the mark. These are called *private*. The algorithms are called *half-public* if a part of the secret key is made public. Other techniques, on the other hand, do not require the unmarked data, nor do they need a key for decoding purposes. They are called *public* because everyone is allowed to access the watermarked data. The algorithm proposed in this paper is half-private.

Digital watermarking algorithms can generally be grouped into three main categories: spatial domain methods [2-7] transform domain methods [8-10] and compressed domain methods [11-13]. Spatial domain methods insert the watermark information in the luminance and/or chrominance values of the pixels. Transform domain methods use the Discrete Cosine Transform (DCT) [9], Discrete Fourier Transform (DFT) [5] or the Wavelet domain [8], [10] for watermark embedding. Watermarking schemes performed in the compressed domain include those for MPEG 1-4 [11-12] and H.26x [13] compressed videos. Such methods usually embed the watermark directly into the VLC (Variable Length Coding) code by modifying the transform domain coefficients [12], [13] or the motion vector information [11].

Watermarking algorithms that embed the watermark in the spatial domain have a relative low arithmetical complexity and are fast, but sometimes not robust enough against specific attacks. In comparison transform domain watermarking techniques are more computationally expensive, but are sometimes more robust to some attacks. Compressed domain methods are fast, robust, but are bound to the compression standard. Any transcoding to a different standard would destroy the watermark.

In this paper, we propose a new approach to spatial domain watermarking for uncompressed videos, robust against most spatial, temporal and compression attacks. The watermark detection process does not require the original video or the original watermark.

The paper is organized as follows: Section 2 introduces our video watermarking method, in section 3 we present the experimental results and section 4 concludes the paper.

## 2. Watermark embedding and extraction

Our watermarking scheme embeds the watermark into the luminance values of the pixels, thus first does a conversion of the RGB (Red Green Blue) color space into the $YC_bC_r$ (Luminance-Chrominance) color space is done:

$$\begin{aligned} Y &= 0.257R + 0.504G + 0.098B + 16 \\ C_b &= -0.148R - 0.291G + 0.439B + 128 \\ C_r &= 0.439R + 0.368G - 0.071B + 128 \end{aligned} \tag{1}$$

After embedding, the video is converted back to the RGB format using Equation 2:

$$\begin{aligned} R &= 1.164(Y-16) + 1.596(C_r - 128) \\ G &= 1.164(Y-16) - 0.813(C_r - 128) - 0.391(C_b - 128) \\ B &= 1.164(Y-16) + 2.018(C_b - 128) \end{aligned} \tag{2}$$

To improve the resilience of the algorithm against different attacks we use three protection mechanics: error correction codes, redundant spatial embedding of one watermark bit into every block of $l \times l$ luminance pixels and redundant temporal embedding of the same watermark in every group of $k$ frames.

The inserted watermark is a binary image of resolution $h \times v$ depending on the size of the original video, the number of redundant frames and the error correction code used.

The algorithm uses a secret key K of 80 bits, 16 for the size of the watermark and 64 bits as seed for the pseudo-random number generator used to produce the code sequence *S*. The pseudo-random binary sequence S is used to spread the power spectrum of the watermark data, thus, increasing its robustness against attacks.

The watermark embedding process, illustrated in Fig. 1, is described in the following steps:

a) The original video is partitioned into groups of $k$ frames.
b) Every frame of the group is converted to the $YC_bC_r$ format as in Equation 1.
c) The binary image matrix is transformed into a binary row vector $w$ of size $P = h \times v$.
d) To protect the watermark against bit errors, a cyclic error correction code ($m$,$n$) with codeword length of $m$ bits and dataword length of $n$ bits is applied to the vector $w$. The size of the resulting watermark vector $w_c$ is:

$$P' = P\frac{m}{n} \tag{3}$$

e) The binary sequence $w_c$ is partitioned into a number of $\frac{C}{k}$ sequences $w_c(j)$ of size $P'\frac{k}{C}$, where $j = \overline{1,\frac{C}{k}}$. The dimensions $h$ and $v$ of the watermark are chosen so that $P'\frac{k}{C}$ is an integer. The same sequence $w_c(j)$ will be inserted into every frame of a group $j$ of $k$ frames.
f) The size $l$ of a square bloc of $l \times l$ luminance values is calculated to embed a bit of the watermark:

$$l = \left[\sqrt{\frac{MNC}{P'k}}\right] \tag{4}$$

where [.] is the integer part operator.
g) A spread-spectrum technique is used to spread the power spectrum of the watermark data, thus, increasing its robustness against attacks. First a binary pseudo-random code sequence $S = \left\{s_r \middle| s_r \in \{0,1\}, r = 0,1,...,l^2\right\}$ of size $l^2$ with equal number of zeros and ones is generated using the Mersenne-Twister algorithm by Nishimura and Matsumoto [14] with the use of the last 64 bits of the secret key $K$ as seed for the generator. This method generates numbers with a period of $(2^{19937} - 1)/2$.
h) For every bit of the watermark $w_c(j)$, the corresponding spread spectrum sequence is:

$$w_{ss} = \begin{cases} [s_1, s_2, ..., s_{l^2}], \text{ if } w_c = 0 \\ [\overline{s}_1, \overline{s}_2, ..., \overline{s}_{l^2}], \text{ if } w_c = 1 \end{cases} \tag{5}$$

i) A sequence $S$ (representing one bit of the original watermark) is embedded in every bloc of $l \times l$ luminance values.

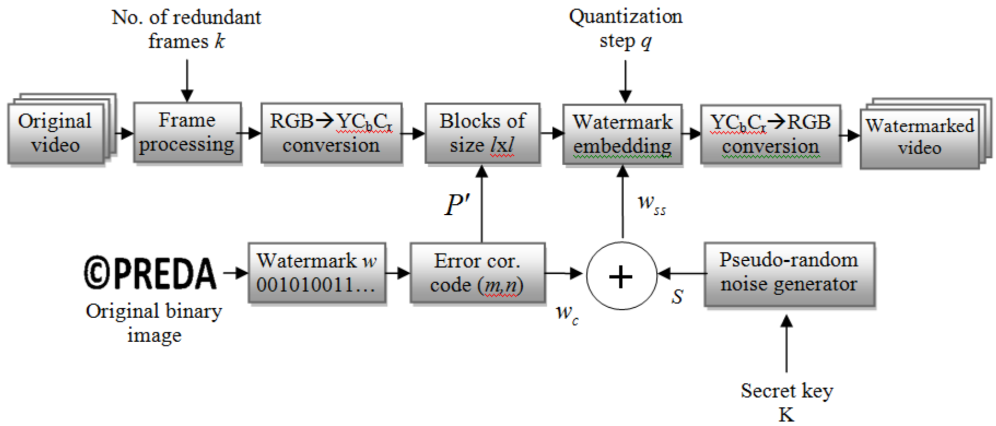


Fig. 1. Block diagram of the proposed watermark encoder

j) A bit of $S$ is embedded into the luminance value of the pixel of the same index by rounding its value to an even or odd quantization level. Rounding to an even quantization level embeds a “0”, while rounding to an odd quantization level embeds a “1”, as shown in Equation 6:

$$L_w(i,j) = \left[\frac{L(i,j)}{2q}\right]\cdot 2q + q\cdot w\cdot sign\left(L(i,j) - \left[\frac{L(i,j)}{2q}\right]\cdot 2q\right), \qquad (6)$$

where $L(i,j)$ is the original luminance value, $L_w(i,j)$ is the watermarked luminance value of the pixel at position $(i,j)$, $q$ is the quantization step size and *sign*() is defined as:

$$sign(x) = \begin{cases} -1, & if\ x \le 0 \\ 1, & if\ x > 0 \end{cases} \qquad (7)$$

*Table 1*

**Embedding a watermark bit into a block of 4x4 luminance pixels**

| Wat. bit | Pseudo-random sequence $S$ | Spread spectrum watermark $w_{ss} = S \oplus w$ | Quant. step | Original luminance block | Watermarked luminance block |
|---|---|---|---|---|---|
| $w$=0 | $\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$ | $\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \end{bmatrix}$ | $q$=4 | $\begin{bmatrix} 224 & 75 & 86 & 20 \\ 62 & 45 & 12 & 123 \\ 45 & 5 & 68 & 74 \\ 145 & 59 & 247 & 23 \end{bmatrix}$ | $\begin{bmatrix} 224 & 76 & 84 & 24 \\ 60 & 48 & 12 & 120 \\ 48 & 4 & 72 & 76 \\ 148 & 60 & 248 & 24 \end{bmatrix}$ |
| $w$=1 | | $\begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$ | | | $\begin{bmatrix} 228 & 72 & 88 & 20 \\ 64 & 44 & 16 & 124 \\ 44 & 8 & 68 & 72 \\ 144 & 56 & 244 & 20 \end{bmatrix}$ |

k) The video is converted back to the RGB format using Equation 2, obtaining the watermark video.

The choice of the quantization step $q$ is a tradeoff between the perceptual quality of the watermarked video ($q$ must have a small value) and the resilience of the watermarking scheme to attacks ($q$ must have a big value). An example of embedding a watermark bit into a block of 4x4 pixels is given in Table 1.

The watermark extraction process, shown in Fig. 2, implies the following steps:

a) The watermarked video is partitioned into groups of $k$ frames.
b) Every frame of the group is converted to the $YC_bC_r$ format using Equation 1.
c) Every luminance frame is partitioned into square blocks of $l \times l$ luminance values.
d) A bit of the spread spectrum sequence $w_{ss}'$ of size $l^2$ is extracted from every luminance value of a block of size $l \times l$ using Equation 8:

$$w' = \mathrm{mod}\,2\left( round\left( \frac{L_w(i,j)}{q} \right) \right) \tag{8}$$

where $w'$ is the extracted watermark bit and mod2($x$) is the modulo2 function.
e) Using the 64 bit seed from the secret key $K$ the binary sequence $S$ is generated locally.
f) The extracted watermark bit of the corresponding block is:

$$w_b' = \begin{cases} 0, & \text{if } \sum_{r=1}^{l^2} \left| w_{ss,r}' - s_r \right| \le \frac{l^2}{2} \\ 1, & \text{if } \sum_{r=1}^{l^2} \left| w_{ss,r}' - s_r \right| > \frac{l^2}{2} \end{cases} \tag{9}$$

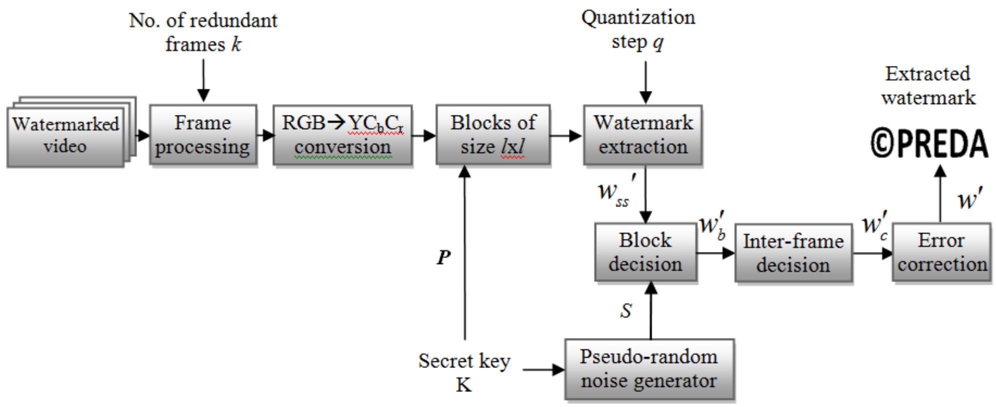


Fig. 2. Block diagram of the proposed watermark decoder

g) A binary sequence $w'_{c,i}(j)$ is extracted from every frame of a group of $k$ frames, where $i=\overline{1,k}$ . The sequence $w'_c(j)$ is computed from $w'_{c,i}(j)$ using Equation 10:

$$w'_c(j)=\begin{cases}0, & \text{dacă} \sum_{i=1}^{k} w'_{c,i}(j) \le \dfrac{k}{2}\\[2ex] 1, & \text{dacă} \sum_{i=1}^{k} w'_{c,i}(j) > \dfrac{k}{2}\end{cases} \quad , \quad j\in\{1,2,\dots,P'\} \tag{10}$$

h) The resulting watermark bitstream $w'_c$ of size *P'* is error corrected and the watermark $w$ of size *P* is obtained.

i) The extracted binary image is obtained by reshaping the vector $w'$ to a matrix of size $h \times v$ .

### 3. Results

The test videos used for our experiments were the first 27 frames of the videos “stefan”, “forman” and “bus” [15] in RGB uncompressed avi format, of resolution 352x288 and frame rate of 30 frames/s. The binary images used as watermarks are shown in Table 2. The choice of the quantization step sizes 2, 4 and 8 was a trade-off between the perceptual quality of the watermarked video and the resilience to attacks. We embedded the watermark with no temporal redundancy and with temporal redundancy in *k*=3 and *k*=9 frames. The error correction code used was a cyclic code (7,4) with codeword length of 7 bits and dataword length of 4 bits. It is a simple code, can correct one error and is useful when the decoding Bit Error Rate (BER) is low (below 14,28%).

*Table 2*

**Binary images used as watermarks**

| Error correction code | No. of redundant frames | Watermark | Watermark resolution |
|---|---|---|---|
| Nu | No |  | 198 x 54 |
| | *k*=3 |  | 108 x 33 |
| | *k*=9 |  | 66 x 18 |
| Cyclic (7,4) | No |  | 144 x 42 |
| | *k*=3 |  | 84 x 24 |
| | *k*=9 |  | 54 x 12 |

First we measured the distortions introduced by the watermark using the mean Peak Signal to Noise Ratio (PSNR) of the three videos after watermark embedding. The results are shown in the third column of Table 2. The PSNR for a video with a number of $C$ frames is calculated using Equation 11.

$$PSNR_{video} = \frac{\sum_{i=1}^{C} PSNR(i)}{C} \tag{11}$$

The resulting PSNR has values between 33 and 44 dB and the watermarked videos appear visually identical to the original ones, except for $q$=8, where a small distortion is visible.

In order to evaluate the robustness and performance of the watermarking algorithm objectively, we have calculated the decoding Bit Error Rate (in percent) and shown the extracted watermark for a set of 7 attacks (Tables 3 and 4).

$$BER = \frac{1}{P}\sum_{j=1}^{P}\left|w_{out}(j) - w_{in}(j)\right| \tag{12}$$

*Table 3*

**Experimental results (first part)**

| Error cor. code | No. red. frames | Quant. step | No attacks | | | Blur | | Gaussian | | Median | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | PSNR | BER (%) | Extracted watermark | BER (%) | Extracted watermark | BER (%) | Extracted watermark | BER (%) | Extracted watermark |
| No | No | 2 | 44.23 | 0.00 | ©PREDA | 11.49 | ©PREDA | 46.35 | | 0.66 | ©PREDA |
| | | 4 | 39.18 | 0.00 | ©PREDA | 8.39 | ©PREDA | 0.00 | ©PREDA | 0.16 | ©PREDA |
| | | 8 | 33.50 | 0.00 | ©PREDA | 1.89 | ©PREDA | 0.00 | ©PREDA | 0.03 | ©PREDA |
| | 3 | 2 | 44.23 | 0.00 | ©PREDA | 7.85 | ©PREDA | 45.51 | | 0.03 | ©PREDA |
| | | 4 | 39.19 | 0.00 | ©PREDA | 6.00 | ©PREDA | 0.00 | ©PREDA | 0.03 | ©PREDA |
| | | 8 | 33.50 | 0.00 | ©PREDA | 0.75 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| | 9 | 2 | 44.22 | 0.00 | ©PREDA | 4.62 | ©PREDA | 41.41 | | 0.00 | ©PREDA |
| | | 4 | 39.18 | 0.00 | ©PREDA | 3.78 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| | | 8 | 33.49 | 0.00 | ©PREDA | 0.42 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| Cyclic (7,4) | No | 2 | 44.23 | 0.00 | ©PREDA | 11.52 | ©PREDA | 47.73 | | 0.05 | ©PREDA |
| | | 4 | 39.18 | 0.00 | ©PREDA | 8.49 | ©PREDA | 0.00 | ©PREDA | 0.03 | ©PREDA |
| | | 8 | 33.50 | 0.00 | ©PREDA | 0.79 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| | 3 | 2 | 44.23 | 0.00 | ©PREDA | 7.63 | ©PREDA | 45.88 | | 0.00 | ©PREDA |
| | | 4 | 39.18 | 0.00 | ©PREDA | 6.15 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| | | 8 | 33.50 | 0.00 | ©PREDA | 0.14 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| | 9 | 2 | 44.23 | 0.00 | ©PREDA | 4.16 | ©PREDA | 44.44 | | 0.00 | ©PREDA |
| | | 4 | 39.18 | 0.00 | ©PREDA | 2.62 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |
| | | 8 | 33.50 | 0.00 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA | 0.00 | ©PREDA |

*Table 4*

**Experimental results (second part)**

| Error cor. code | No. red. frames | Quant. step | Salt and pepper | | Fr. averaging (3 frames) | | JPEG Q=80 | | MPEG-2 4 Mbps | | MPEG-2 2 Mbps | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | BER (%) | Extracted watermark | BER (%) | Extracted watermark | BER (%) | Extracted watermark | BER (%) | Extracted watermark | BER (%) | Extracted watermark |
| No | No | 2 | 0.00 | | 5.39 | | 42.70 | | 40.33 | | | |
| | | 4 | 0.00 | | 3.73 | | 22.09 | | 24.37 | | 47.37 | |
| | | 8 | 0.00 | | 3.70 | | 0.02 | | 10.13 | | 26.68 | |
| | 3 | 2 | 0.00 | | 0.00 | | 39.42 | | 34.15 | | 45.51 | |
| | | 4 | 0.00 | | 0.00 | | 17.79 | | 15.29 | | 34.99 | |
| | | 8 | 0.00 | | 0.00 | | 0.00 | | 3.51 | | 15.99 | |
| | 9 | 2 | 0.00 | | 0.00 | | 35.10 | | 30.13 | | 42.93 | |
| | | 4 | 0.00 | | 0.00 | | 15.91 | | 7.24 | | 29.12 | |
| | | 8 | 0.00 | | 0.00 | | 0.00 | | 0.59 | | 6.90 | |
| Cyclic (7,4) | No | 2 | 0.00 | | 5.13 | | 44.74 | | 40.63 | | 47.60 | |
| | | 4 | 0.00 | | 3.70 | | 21.02 | | 24.45 | | 41.63 | |
| | | 8 | 0.00 | | 3.69 | | 0.00 | | 9.87 | | 27.41 | |
| | 3 | 2 | 0.00 | | 0.00 | | 39.43 | | 34.23 | | 47.92 | |
| | | 4 | 0.00 | | 0.00 | | 16.22 | | 13.69 | | 34.28 | |
| | | 8 | 0.00 | | 0.00 | | 0.00 | | 2.13 | | 15.18 | |
| | 9 | 2 | 0.00 | | 0.00 | | 34.10 | | 31.02 | | 41.67 | |
| | | 4 | 0.00 | | 0.00 | | 16.82 | | 8.18 | | 31.79 | |
| | | 8 | 0.00 | | 0.00 | | 0.00 | | 0.15 | | 6.94 | |

In equation 12 $w_{out}$ is the extracted watermark, $w_{in}$ is the original watermark and $P$ is the size of the watermark. In Tables 3 and 4 PSNR and BER are mean values for the 3 test videos, but, because of space requirements, we are showing the extracted watermark only for the “stefan” video.

The fifth and sixth column of Table x show the decoding BER and the extracted watermark images in absence of any attack. The extracted watermarks are identical to the original ones.

We used the following seven attacks to test the robustness of our watermarking scheme:

- Blurring using blocks of 2x2 pixels
- Adding gaussian noise of mean 0 and variance 0,05%
- Median filtering using a 3x3 pixel neighborhood
- Adding “salt and pepper” noise with density d=0,05%
- Frame averaging of 10% of the frames, where the current frame

$$f(i) = \frac{f(i-1) + f(i) + f(i+1)}{3} \tag{13}$$

- JPEG compression of every frame with quality factor Q=80
- MPEG-2 compression at 4 and 2 Mbps

Fig. 3 shows the decoding BER for every attack for a quantization step size $q$=2 and without any error correcting code and Fig. 4 and 5 show the decoding BER for the quantization step size $q$=4 and $q$=8 with the use of the cyclic error correction code (7,4). For $q$=2 the BER values are relatively high and the use of error correction code brings no improvement.

The experimental results show, that our watermarking scheme is robust to most of the attacks. Some detection problems occur after the gaussian noise attack, but only for the smallest quantization step, $q$=2, and after MPEG-2 compression at 2 Mbps.

Increasing the size of the quantization step $q$ the resilience of the scheme against all attacks is improved, but the perceptual quality of the watermarked video is decreased. The redundant embedding of the same watermark in 3 or 5 video frames improves the resilience of the algorithm against all attacks.

## 4. Conclusions

In this paper, we proposed a novel video watermarking technique based on the quantization of the luminance values of pixel blocks. The algorithm uses error correction codes to protect the inserted watermark and temporal redundancy to embed the same watermark in different frames of the video. The watermarks used

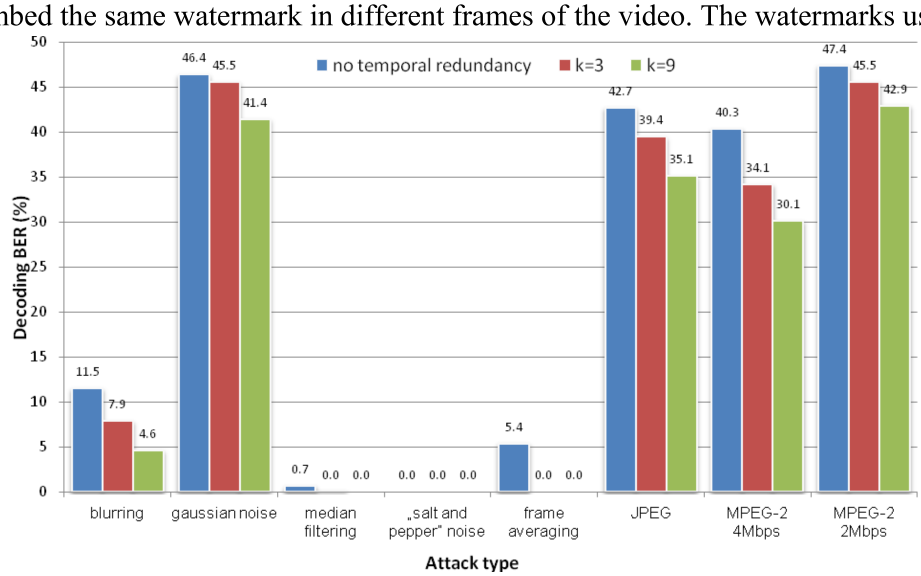


Fig. 3. Decoding BER (%) for all attacks using a quantization step size $q$=2 and no error correction code

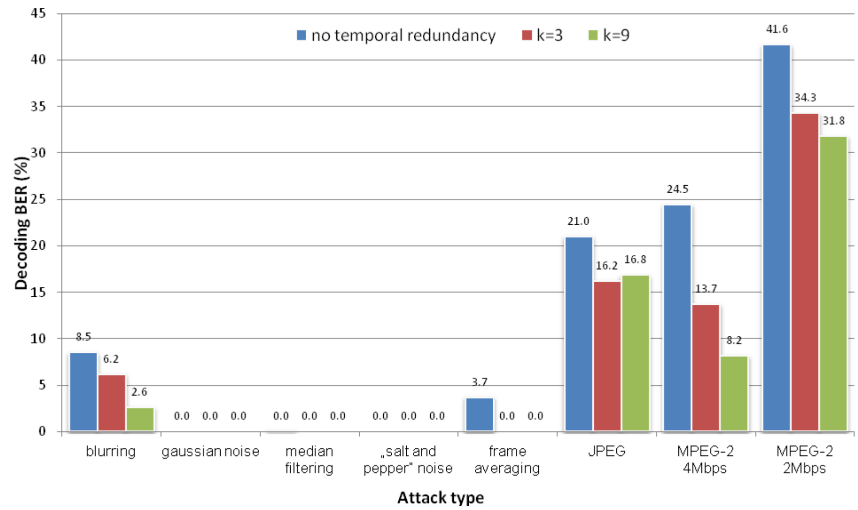


Fig. 4. Decoding BER (%) for all attacks using a quantization step size $q$=4 and a cyclic error correction code (7,4)

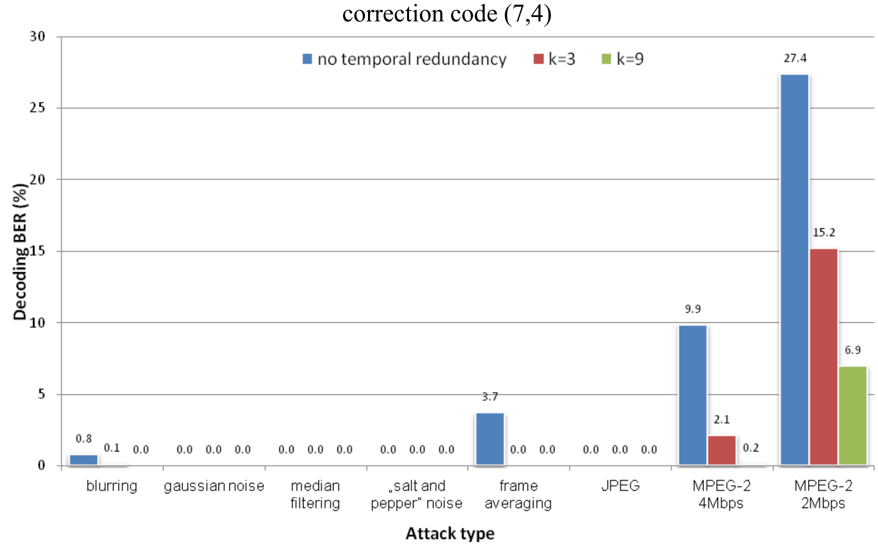


Fig. 5. Decoding BER (%) for all attacks using a quantization step size $q$=8 and a cyclic error correction code (7,4)

were binary images containing the name of the author. The advantage of using binary images instead of pseudo-random noise is that even a watermark extracted with a high decoding BER can be visually identified. The perceptual quality of the watermarked videos is very good for the quantization step sizes of 2 and 4 (PSNR above 39dB) and acceptable for a quantization step size of 8 (PSNR=33,5dB). The proposed technique also achieves good resilience against a seven different attacks in the spatial domain (blurring, gaussian noise, median filtering, “salt and pepper” noise), temporal domain (frame averaging) and compressed domain (JPEG, MPEG-2 compression).

**Acknowledgments**
This research was funded by UEFISCSU, 2011, D.N. Vizireanu.

## REFERENCES

[1] *D. Kundur, K. Su, D. Hatzinakos*, “Digital Video Watermarking: Techniques, Technology and Trends”, in Intelligent Watermarking Techniques, Chapter 10, World Scientific Publishing Company, 2004, pp. 265-314

[2] *M. Kutter, F.Jordan, F.Bossen*, Digital watermarking of color images using amplitude modulation, in Journal of Electronic Imaging, **vol. 7**, no. 2, 1998, pp.326-332

[3] *P.T.Yu, H.H.Tsai, J.S.Lin*. Digital watermarking based on neural networks for color images, in Signal Processing, **vol. 81**, Issue: 3, March, 2001, pp. 663-671

[4] *R. Lancini, F. Mapelli, S. Tubaro*, A robust video watermarking technique in the spatial domain, in Proc. of the 8th IEEE Int. Symposium on Video/Image Processing and Multimedia Communications, June 2002, pp. 251-256

[5] *Y. Yuan and C.T. Li*, Fragile Watermarking Scheme Exploiting Non-deterministic Block-wise Dependency, in Proc. of the IAPR Int. Conf. on Pattern Recognition, **vol. 4**, Cambridge, UK, 2004, pp. 849-852

[6] *P.S. Huang, C.-S. Chiang, C.-P. Chang, T.-M. Tu,* Robust spatial watermarking technique for colour images via direct saturation adjustment, in IEEE Proc.- Vis. Image Signal Process., **vol. 152**, no. 5, October 2005

[7] *D. P. Mukherjee, S. Maitra, S. T. Acton*, Spatial Domain Digital Watermarking of Multimedia Objects for Buyer Authentication, in IEEE Transactions on Multimedia, **vol. 6**, No. 1, 2004, pp. 1-19

[8] *X. Kang, J. Huang, Y. Q Shi, Y. Lin*, A DWT-DFT composite watermarking scheme robust to both affine transform and JPEG compression, IEEE Transactions on Circuits and Systems for Video Technology, **vol. 13**, Issue 8, Aug, 2003, pp.776-786

[9] *Lian-Shan Liu, Ren-Hou Li, Qi Gao*, A robust video watermarking scheme based on DCT, in Proceeding of the IEEE International Conference on Machine Learning and Cybernetics, **vol. 8**, Aug. 2005, pp.5176-5180

[10] *Y. Li, X. Gao, H. Ji*, A 3D wavelet based spatial-temporal approach for video watermarking, in Proc. Of 5th Int. Conf on Computational Intell. And Multimedia Applications (ICCIMA), Sept. 2003, pp.260-265

[11] *D. Y. Jun, Z. L. He , Y. Y. Xian*, A new method of MPEG video watermarking technology, in Proceedings of t he IEEE ICCT, Beijing, 2003, pp. 1845-1847

[12] *S. Biswas, S.R. Das, E.M. Petriu*, An adaptive compressed MPEG-2 video watermarking scheme, in IEEE Transactions on Instrumentation and Measurement, **vol. 54**, no. 5, 2005, pp. 1853-1861

[13] *J. Zhang, A. Ho, G. Qiu, P. Marziliano*, Robust video watermarking of H.264/AVC, in IEEE Transactions on Circuits and System-II: Express Briefs, **vol. 54**, February 2007, pp. 205–209

[14] *M. Matsumoto, T. Nishimura*, Mersenne Twister: A 623-Dimensionally Equidistributed Uniform Pseudorandom Number Generator, in ACM Transactions on Modeling and Computer Simulation, **vol. 8(1)**, 1998, pp. 3-30

[15] *http://trace.eas.asu.edu/*