vol. 66, no. 3, May 1998
pp. 283-301

Watermarking of Uncompressed and Compressed Video

Frank Hartung and Bernd Girod

Telecommunications Institute I
University of Erlangen-Nuremberg
Cauerstrasse 7, 91058 Erlangen, Germany
Phone +49 9131 85 7101
Fax +49 9131 85 8849
{hartung,girod}@nt.e-technik.uni-erlangen.de

In this paper, methods for embedding additive digital watermarks into uncompressed and compressed video sequences are presented. The basic principle borrows from spread spectrum communications. It consists of addition of an encrypted, pseudo-noise signal to the video that is invisible, statistically unobtrusive, and robust against manipulations. For practical applications, watermarking schemes operating on compressed video are desirable. A method for watermarking of MPEG-2 encoded video is presented. The scheme is a compatible extension of the scheme operating on uncompressed video. The watermark is generated exactly in the same manner as for uncompressed video, transformed using the discrete cosine transform (DCT) and embedded into the MPEG-2 bitstream without increasing the bit-rate. The watermark can be retrieved from the decoded video and without knowledge of the original, unwatermarked video. Although an existing MPEG-2 bitstream is partly altered, the scheme avoids visible artifacts by addition of a drift compensation signal. The proposed method is robust and of much lower complexity than a complete decoding process followed by watermarking in the pixel domain and re-encoding. Fast implementations exist which have a complexity comparable to a video decoder. Experimental results are given. The scheme is also applicable to other hybrid transform coding schemes like MPEG-1, MPEG-4, H.261, and H.263.

Key words: digital watermarking, copyright protection, compressed video, MPEG-2, MPEG-4, DVD

Preprint submitted to Elsevier Preprint

1 Introduction

With the advent of digital video, issues of copyright protection have become more important, since the duplication of digital video signals does not result in the inherent decrease in quality suffered by analog video. A method of copyright protection is the addition of a "watermark" to the video signal. The watermark is a digital code embedded in the video which can be used for the embedded transmission of binary information and which typically indicates the copyright owner. If different watermarks are applied to individual copies of the video, watermarking can also be used to indicate the identity of the legal receiver of each copy. This allows illegally reproduced copies to be traced back to the receiver from which they originated, as shown in Fig. 1.

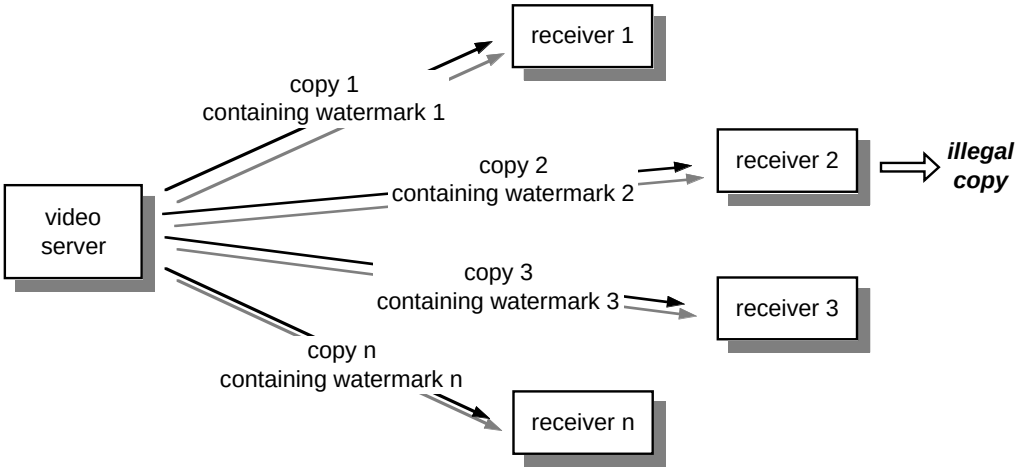

Fig. 1. Principle of individual video watermarking.

The idea of watermarking is in fact very old: methods for embedding invisible information which can be used to distinguish different copies of written or printed documents have been used by secret services and agents for centuries, only that this art was called "steganography". For formatted digital text documents, the idea was revived in 1994 by Brassil et al. [1]. Even before, Caronni had worked on first methods for embedding invisible information into digital images [2,3]. Subsequent publications deal with embedding watermarks (in some publications also called "label" or "signature") into audio [4], uncompressed digital images [5–10], JPEG-compressed digital images [5], uncompressed digital video [11], compressed digital video [11–13], and 3D polygonal models [14]. In some publications it is also proposed to place *visible* watermarks in images [15]. A very complete overview over the development of watermarking is given in [16].

For digital watermarking of video, a number of different characteristics of the watermarking process and the watermark are desirable. These requirements

are

- **Invisibility:** The digital watermark embedded into the video data should be invisible to the human observer.
- **Security:** Unauthorized removal of the watermark must be impossible once it has been embedded, even if the basic scheme used for watermarking is known, as long as the exact parameters are unknown.
- **Robustness:** It should be impossible to manipulate the watermark by intentional or unintentional operations on the uncompressed or compressed video without, at the same time, degrading the perceived quality of the video to the point of significantly reducing its commercial value. Such operations are, for example, addition of signals, filtering, cropping, encoding, or analog recording and playback.
- **Complexity:** Watermarking and watermark retrieval should in principle have low complexity. Different applications do, however, pose different requirements on complexity. If watermarking is used for audit trail, each receiver has to retrieve the watermark, and watermark retrieval should be easy. If watermarking is used for embedding individual receiver identity labels, watermarking is performed on a large number of distributed video sequences, while watermark retrieval occurs only in cases where possible copyright violations have to be investigated. While the retrieval operation may be more complex in order to account for all possible kinds of attacks on the watermark, watermarking should be of low complexity in such cases.
- **Compressed domain processing:** It can be assumed that the distributor or broadcaster of digital video will usually store the video in compressed format, for example on a video-on-demand server, or a World Wide Web server. Referring to the above complexity requirement, it should be possible to incorporate the watermark into the compressed video (the bitstream), because
  - it is too complex and not feasible to decode and re-encode the video for watermarking
  - the quality of decoded and re-encoded video can in general not be guaranteed.
- **Constant bit-rate:** Watermarking in the bitstream domain should not increase the bit-rate, at least for constant bit-rate applications where transmission channel bandwidth has to be obeyed.
- **Interoperability:** Even though many applications call for watermarking of compressed video, it would be a desirable property if uncompressed video could compatibly be watermarked without having to encode it first, as shown in Fig. 2.

Applications for watermarking methods with the listed specifications arise where multiple copies of video data are distributed and where it is feasible to process them in order to embed watermarks. Examples are point-to-point distribution of compressed digital video over the Internet, production of digital

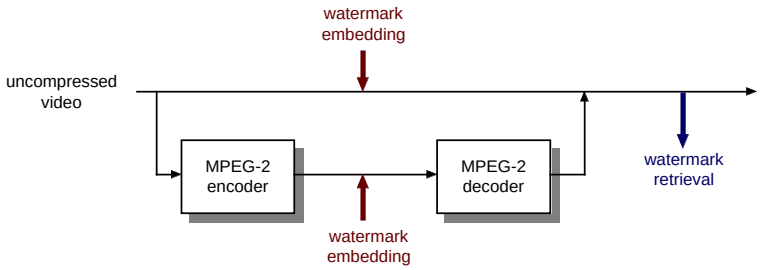

Fig. 2. Interoperability of watermarking in the uncoded and coded domain.

video tapes and digital video discs (DVD) where each copy can get an own identity [17], or pay-per-view television broadcasting where the video signal is individually watermarked for each receiver [18]. In the latter application, the receiver would typically have a set-top box or similar apparatus which incorporates conditional access (decryption) and video decoding. For this type of applications, watermarking might be carried out in the set-top box at the receiver, in order to move the complexity connected with watermarking from a central video server to distributed set-top boxes, as shown in Figure 3.

- - -

2 Spread Spectrum Watermarking of Uncompressed Video

Spread spectrum communication schemes transmit a narrow-band signal via a wide-band channel by frequency spreading [19]. For watermarking, ideas from spread spectrum communications are highly applicable: a narrow-band signal (the watermark) has to be transmitted via a wide-band channel with interference (the image or video signal). Specifically, the idea of direct sequence spread spectrum communication can be adopted for watermarking of video in a similar fashion as in [8], as explained in the following.

2.1 Embedding of the Watermark

Often, a video sequence is regarded as a three-dimensional signal with two dimensions in space and one dimension in time. For our purposes however, we regard the video signal as a one-dimensional signal acquired by line-scanning, as depicted in Fig. 4.

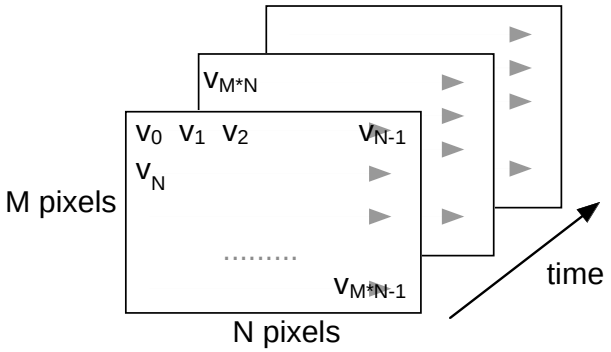

Fig. 4. Line scan of a video signal

Let us denote

$$a_j, \quad a_j \in \{-1, 1\}, \quad j \in \mathbf{N} \tag{1}$$

a sequence of watermark bits that has to be embedded into the video stream. This discrete signal is spread by a large factor cr, called the chip-rate, to obtain the spread sequence

$$b_i = a_j, \quad j \cdot cr \leq i < (j+1) \cdot cr, \quad i \in \mathbf{N} \tag{2}$$

The purpose of spreading is to add redundancy by embedding one bit of information into cr pixels of the video signal. The spread sequence b_i is amplified

with a locally adjustable amplitude factor $\alpha_i \geq 0$ and is then modulated by a binary pseudo-noise sequence

$$p_i, \quad p_i \in \{-1, 1\}, \quad i \in \mathbf{N} \tag{3}$$

which serves for frequency spreading [19]. The modulated signal, i.e. the spread spectrum watermark

$$w_i = \alpha_i \cdot b_i \cdot p_i, \quad i \in \mathbf{N} \tag{4}$$

is added to the line-scanned digital video signal v_i yielding the watermarked video signal

$$\tilde{v}_i = v_i + \alpha_i \cdot b_i \cdot p_i, \quad i \in \mathbf{N} \tag{5}$$

which must be re-arranged into a matrix for display. Due to the noisy nature of the pseudo-noise signal p_i, the watermark signal w_i is also a noise-like signal and thus difficult to detect, locate, and manipulate. Figure 5 visualizes the principle of watermark embedding with help of an example.

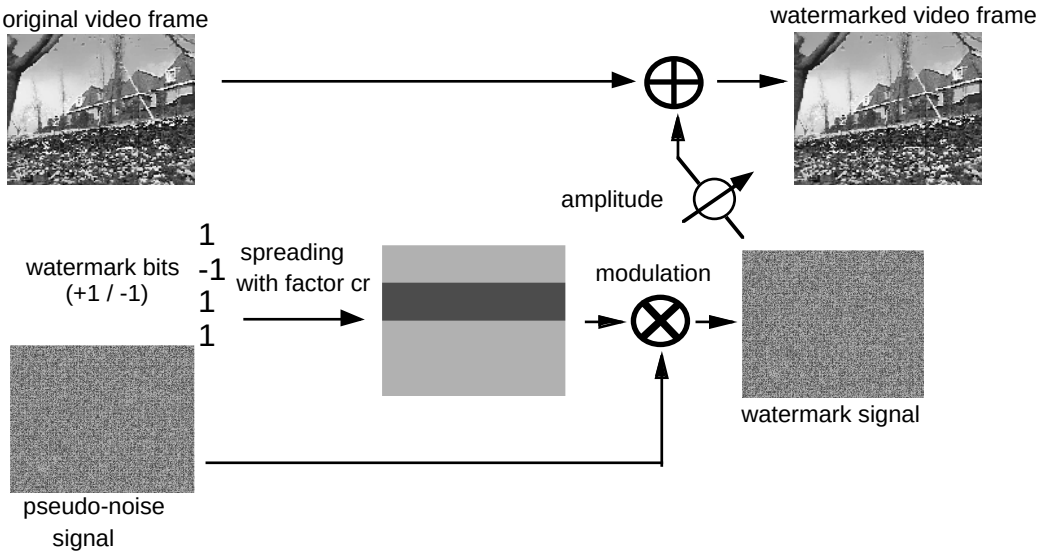

Fig. 5. Visualization of watermark embedding

For simplicity, a binary pseudo-noise sequences is assumed in (3). There exist infinitely many such sequences. Such sequences can for example be generated by feed-back shift registers producing m-sequences, any other random number generator, or by chaotic physical processes [20]. Since the pseudo-noise signal is the secret key for embedding and retrieval of the watermark, and for security reasons, sequences should be used that are not easy to guess (like short m-sequences are).

Non-binary, e.g. Gaussian, pseudo-noise sequences are also possible without other modifications of the scheme.

Different pseudo-noise sequences are in general orthogonal to each other and do not significantly interfere [21]. Thus, several watermarks can be superimposed, if different pseudo-noise sequences are used for modulation. They can be retrieved in arbitrary order and independently from each other.

The amplitude factor α_i may be varied according to local properties of the video signal and can be used to exploit spatial and temporal masking phenomena of the human visual system [22] such that the amplitude of the watermark is locally as large as possible without becoming visible. More watermark information can be embedded in areas of the video frames where it is less visible, for example in areas containing fine detail, or in the video frames following scene cuts. In image watermarking, such psychovisual models are often adopted in order to accommodate a maximum of information into a single picture [23,24]. For video watermarking, the watermark data rate that can be achieved with a constant, low amplitude is high enough for many applications. In this cases, it is not necessary or appropriate to use a psychovisual model, since such models are often prohibitively complex.

2.2 Retrieval of the Watermark

Authorized recovery of the hidden information is easily accomplished, even without knowledge of the original, unwatermarked signal, by means of a correlation receiver. Prior to the correlation step, the input signal, i.e. the watermarked video sequence $\tilde{v}$, is highpass filtered, yielding a filtered watermarked video signal $\bar{\tilde{v}}$. The purpose of the filtering operation is to separate and remove major components of the video signal itself. The filter may be a one-dimensional or two-dimensional adaptive or non-adaptive filter. We have, for example, used a non-adaptive 3×3 highpass filter. Filtering is not necessary, but improves the performance of the overall watermarking system, because it reduces cross-talk between watermark signal and video signal. The second step is demodulation, i.e. multiplication of the filtered watermarked video signal with the same pseudo-noise signal p_i that was used for embedding, followed by summation over the window for each embedded bit, yielding the correlation sum s_j for the j'th information bit:

$$s_j = \sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} p_i \cdot \bar{\tilde{v}}_i = \underbrace{\sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} p_i \cdot \bar{v}_i}_{\Sigma_1} + \underbrace{\sum_{i=j\cdot cr}^{(j+1)\cdot cr-1} p_i \cdot \overline{p_i \cdot \alpha_i \cdot b_i}}_{\Sigma_2} \qquad (6)$$

where the two terms Σ_1 and Σ_2 describe the contributions to the correlation sum from the filtered video signal and the filtered watermark signal, respectively. Before (6) is examined in more detail in section 2.3, let us first assume that the sum Σ_1 is zero, that means the video signal has been filtered out in $\bar{\tilde{v}}$, and that $\overline{p_i \cdot \alpha_i \cdot b_i} \approx p_i \cdot \alpha_i \cdot b_i$, that means that the highpass filtering has negligible influence on the white pseudo-noise watermark signal. Under these assumptions, the correlation sum becomes

$$s_j = \Sigma_1 + \Sigma_2 \approx \sum_{i=j \cdot cr}^{(j+1) \cdot cr - 1} p_i^2 \cdot \alpha_i \cdot b_i = a_j \cdot \sigma_p^2 \cdot cr \cdot mean(\alpha_i), \qquad (7)$$

where σ_p^2 is the variance of the pseudo-noise sequence. Thus, the sign of the correlation sum is just the embedded information bit

$$sign(s_j) = sign(a_j \cdot \underbrace{\sigma_p^2 \cdot cr \cdot mean(\alpha_i)}_{>0}) = sign(a_j) = a_j \qquad (8)$$

so that the embedded information can be retrieved losslessly. (8) means that the transmitted bit was a $+1$ if the correlation between the video signal with embedded watermark containing the current bit and the pseudo-noise signal is positive. If the correlation is negative, the transmitted bit was a -1.

If the wrong pseudo-noise sequence is used, or if it is not in synchronization with the pseudo-noise sequence as used for embedding, the scheme does not work, and the recovered bits are random. Thus, the watermark decoder has to know the pseudo-noise sequence and its possible shift. If the pseudo-noise sequence is known, but its shift is unknown, synchronization can be found by means of a sliding correlator: all possible shifts are experimentally applied, and the right shift is found, if the modified correlation sum is significantly larger than for all other shifts. However, finding the correlation is cumbersome and complex, especially for pseudo-noise sequences with a very large cycle.

The recovery of the embedded information, as described, does not require the unwatermarked original signal. However, the recovery of the embedded information is more robust, if the original, unwatermarked signal is known, and can be subtracted before demodulation instead of the filtering operation, because the subtraction removes all interference between the video signal itself and the embedded watermark.

type of filter	mean and variance of $\bar{v}$
$\bar{v} = 0$ (video signal removed)	$\mu_{\bar{v}} = 0$ $\sigma^2_{\bar{v}} = 0$
$\bar{v}$ obtained by 3×3 highpass filtering	$\mu_{\bar{v}} = 0$ $\sigma^2_{\bar{v}} = \ldots 900$
$\bar{v} = v$ (no filtering)	$\mu_{\bar{v}} = 127.5$ $\sigma^2_{\bar{v}} = 5461.2$

Table 1
Typical variances for the filtered video signal.

2.3 Performance and Robustness

While the assumption that the highpass filtering has small influence on the pseudo-noise watermark signal is true for appropriate filters with sufficiently narrow stop-band, the assumption that Σ_1 is zero is in general not true, because some energy of the video signal remains in the filtered watermarked video signal. Thus, we have to consider the influence of Σ_1 which leads to occasional bit errors. A bit error occurs if $sign(\Sigma_1 + \Sigma_2) \neq sign(\Sigma_2)$, that is, if $sign(\Sigma_1) \neq sign(\Sigma_2)$ and $\mid \Sigma_1 \mid > \mid \Sigma_2 \mid$. In the following, the probability for this to happen is estimated, depending on the parameters used (chip-rate cr, amplitude amplification α_i, variance of the spreading pseudo-noise function σ_p^2), the properties of the filtered video signal $\bar{v}$ and the properties of video signals in general. The properties of the filtered video signal $\bar{v}$ depend on the filter used. Typical values for mean and variance of $\bar{v}$ are listed in Table 1. In the sum Σ_1 describing the distortion term of the correlation sum, the filtered video signal is multiplied with the PN sequence p having mean $\mu_p = 0$ and variance σ_p^2. The product has mean $\mu_{p\bar{v}} = 0$ and variance $\sigma^2_{p\bar{v}} = \sigma_p^2 \cdot (\sigma^2_{\bar{v}} + \mu^2_{\bar{v}})$. In the sum Σ_1, the product $p\bar{v}$ is summed cr times. Thus, according to the central limit theorem [25], the probability density function of the sum approaches a normal distribution with mean $\mu_{\Sigma_1} = cr\ \mu_{p\bar{v}} = 0$ and variance $\sigma^2_{\Sigma_1} = cr \cdot \sigma^2_{p\bar{v}} = cr \cdot \sigma_p^2 \cdot (\sigma^2_{\bar{v}} + \mu^2_{\bar{v}})$. A bit error occurs if the current information bit is a $+1$ and $\Sigma_1 < -\sigma_p^2 \cdot cr \cdot mean(\alpha_i)$, or if the current information bit is a -1 and $\Sigma_1 > \sigma_p^2 \cdot cr \cdot mean(\alpha_i)$. Since Σ_1 can be described by a normal distribution, the bit error rate is

$$\mathrm{BER} = probability(\Sigma_1 > \sigma_p^2 \cdot cr \cdot mean(\alpha_i)) \tag{9}$$

$$= \frac{1}{\sqrt{2\pi}\sigma_{\Sigma_1}} \int_{\sigma_p^2 \cdot cr \cdot mean(\alpha_i)}^{\infty} \exp\left(\frac{-t^2}{2\sigma^2_{\Sigma_1}}\right) dt \tag{10}$$

chip rate cr	amplification $mean(\alpha_i)$	PN variance σ_p^2	used filter	estimated BER	measured BER
1000	1	1	no filtering	0.415	0.412
10000	3	1	no filtering	0.021	0.018
50000	4	1	no filtering	7.2×10^{-10}	$\approx 0^{\dagger}$
1000	1	1	3×3 HP filter	0.146	4.8×10^{-2}
1000	2	1	3×3 HP filter	1.8×10^{-2}	8.1×10^{-3}
1000	3	1	3×3 HP filter	7.8×10^{-4}	5.5×10^{-4}
5000	1	1	3×3 HP filter	9.2×10^{-3}	5.1×10^{-3}
5000	2	1	3×3 HP filter	1.2×10^{-6}	$\approx 0^{\dagger}$
10000	3	1	3×3 HP filter	7.6×10^{-24}	$\approx 0^{\dagger}$
10000	4	1	3×3 HP filter	7.4×10^{-41}	$\approx 0^{\dagger}$
> 0	> 0	> 0	$\bar{v} = 0$ (v removed)	0	$\approx 0^{\dagger}$

Table 2
Examples of estimated and measured bit error rates for embedded watermarks. (†: no bit errors observed for a watermark of length 10000 bit.)

$$= \frac{1}{\sqrt{2\pi}\sigma_{\Sigma_1}} \sqrt{\frac{\pi}{2}} \sigma_{\Sigma_1} \operatorname{erfc}\left(\frac{\sigma_p^2 \cdot cr \cdot mean(\alpha_i)}{\sqrt{2} \cdot \sigma_{\Sigma_1}}\right) \tag{11}$$

and finally

$$\mathrm{BER} = \frac{1}{2} \operatorname{erfc}\left(\frac{\sigma_p \cdot \sqrt{cr} \cdot mean(\alpha_i)}{\sqrt{2} \cdot \sqrt{\sigma_{\bar{v}}^2 + \mu_{\bar{v}}^2}}\right). \tag{12}$$

At the same time, the data rate R_{WM} for the watermark is

$$R_{WM} = \frac{\text{number of luminance pixels per second}}{cr} \tag{13}$$

Increase of chip-rate cr, average amplitude $mean(\alpha_i)$, or variance of the pseudo-noise signal σ_p decrease the bit error rate; using a poor filter which does not remove the video signal from the watermarked video signal very well increases the bit error rate. Table 2 gives a few examples for parameters used and the resulting bit error rates. It can be seen that bit error rates below 10^{-10} can easily be accomplished. A good choice of parameters is for example $cr = 2400$, $mean(\alpha_i) = 3$, $\sigma_p^2 = 1$ and using a 3×3 highpass filter, resulting in a bit error rate of approximately 5×10^{-7} and a watermark data rate of $R_{WM} = 528$ byte/s for NTSC video. The theoretically acquired bit error rate estimates were

confirmed by experimental results. The measured bit error rates are displayed in the right column of Table 2. However, these estimations and experiments are valid only for embedding of watermarks into uncompressed video, and do not include the effects of malicious attacks on the watermarks. If a desired bit error rate has to be maintained in the presence of attacks, the argument of (12) has to be increased by a heuristic safety factor which depends on the type and severeness of attacks the watermark has to resist. This is easily accomplished by increasing chip-rate cr and/or amplitude α_i. However, increase of cr reduces the data rate for the watermark information, i.e. the number of information bits per second that can be embedded into the video. If this done appropriately, the embedded watermark is robust against intentional and unintentional operations on the watermarked signal, e.g. filtering, addition of an offset, addition of white, colored or impulse noise, cropping, quantization in spatial or frequency domain, compression, and others. The robustness is a consequence of the fact that each bit of information to be embedded is spread over a large number of pixels.

Figure 6 gives a visual impression of some example attacks on a watermarked video that the watermark survived. The depicted attacks are attacks on the

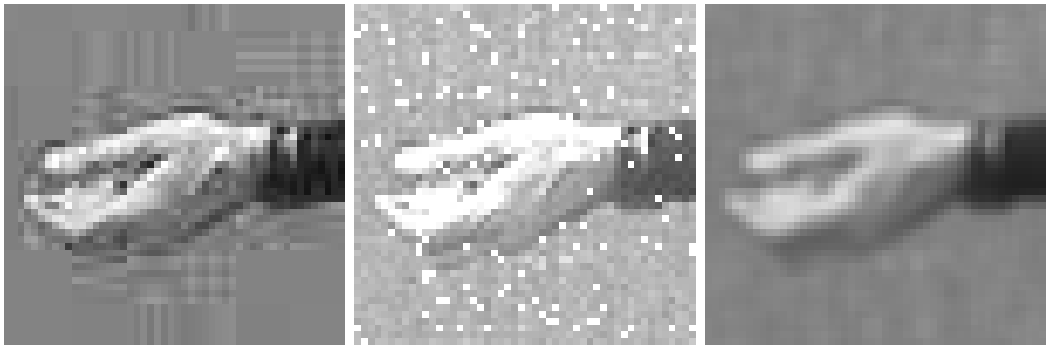

Fig. 6. Details of a video sequence attacked by different approaches. The watermark survived all those attacks. Left: blockwise DCT compression; middle: addition of Gaussian and impulse noise and an offset, right: lowpass filtering.

pixel level. More cumbersome to circumvent are attacks which try to destroy correlation between the embedded watermark and the original pseudo-noise sequence, for example by removing single pixels, lines or frames from the video, by rotation or by affine transformation of the video frames. In those cases, the retrieval of the watermark looses synchronization. Countermeasures are for example tracking of the correlation, and re-synchronization by means of a sliding correlator in cases of synchronicity loss, or blockwise hierarchical correlation similar to hierarchical block-matching [26]. This may be complex in real applications, but since this type of attack only attempts to hide the watermark, and not to remove or destroy it, it is always possible to re-synchronize and retrieve the watermark, if enough effort is put into it.

With decreasing chip-rate, single bits of the watermark information may eventually be decoded incorrectly, according to (12). The use of an error-correcting code in order to protect the information bits can help to lower the minimum chip-rate, and to increase the bit-rate of the watermark information. Additional gain is possible, if an error correcting code with soft-bit decision [27] is used, rather than the hard-bit decision of (8).

3 Spread Spectrum Watermarking of Compressed Video

In practical video storage and distribution systems, the video sequences are stored and transmitted in compressed format. In this case, if different copies have to be watermarked with individual watermarks, decoding, watermarking in the pixel domain and re-encoding is not feasible, for the reasons stated in section 1. Thus, if different watermarks have to be embedded into different copies of a video sequence, it must obviously be done by operations on the compressed video. After the watermark is embedded, the watermarked video is distributed and possibly decoded. The watermark must persist in the decoded video and must be retrievable from the decoded video. In the following, a scheme for watermarking of compressed video is presented that fulfills the mentioned requirements. The scheme is fully compatible with the scheme of section 2 operating in the pixel domain. This means, a watermark can either be embedded into the uncompressed (with the scheme of section 2) or compressed video (with the scheme of section 3), and can be retrieved from the decompressed video (with the scheme of section 2.2). The scheme for compressed-domain embedding can be also applied to embed any other additive watermark signal into compressed video.

3.1 Principle

All current international standards for video compression, namely MPEG-1, MPEG-2, the baseline mode of MPEG-4, ITU-T H.261, and ITU-T H.263, are hybrid coding schemes. Such schemes are based on the principles of motion compensated prediction and block-based transform coding. The transform used is always the discrete cosine transform (DCT), except for non-rectangular border blocks of MPEG-4 video objects, where it is its derivative, the shape-adaptive DCT (SA-DCT) [28]. In the following, we refer specifically to MPEG-2 compression. However, the ideas presented here apply to all other hybrid coding schemes as well. Figure 7 shows a generic block diagram of a hybrid coding scheme. Intra-coded frames (in MPEG-2 terminology: I-frames) are split into blocks of 8 by 8 pixels which are compressed using the DCT, quantization (Q), zig-zag-scan, run-level-coding and entropy coding (VLC).

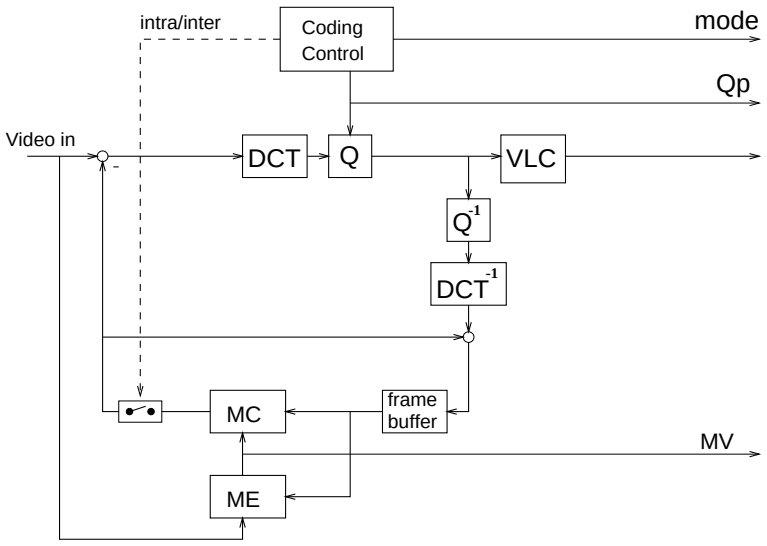

Fig. 7. Hybrid video coding scheme.

Inter-coded frames (in MPEG-2 terminology: P- or B-frames) are subject to motion compensation by subtracting a motion compensated prediction. The residual prediction error signal frames are split into blocks of 8 by 8 pixels which are compressed in the same way as blocks from inter-frames. Fig. 8 depicts the procedure for encoding of a single 8×8-block which is, in the bitstream, represented as a series of Huffman codewords.

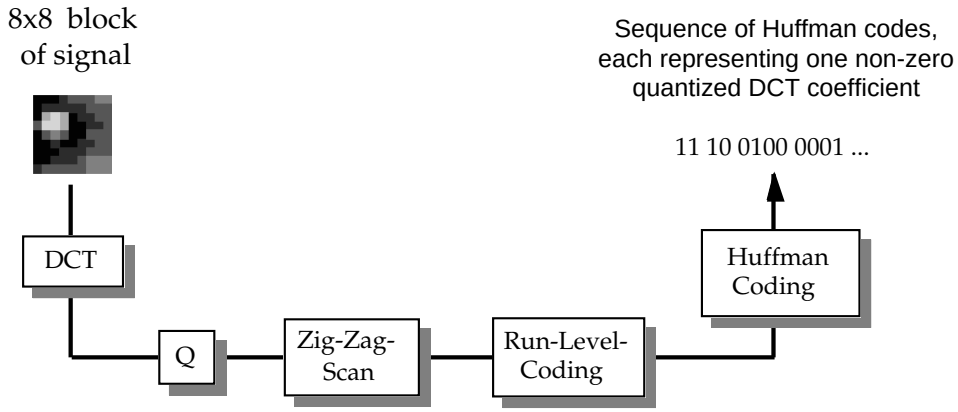

Fig. 8. DCT encoding of one 8x8 pixel block.

The basic idea for watermarking of MPEG-2 coded video is

(i) generating a watermark signal for each frame of the video sequence exactly in the same manner as it is done in the pixel domain (see section 2).
(ii) arranging the watermark signal into a two-dimensional signal having the same dimensions as the video frames

Fig. 9. Generic scheme for watermarking of compressed video

incoming MPEG-2 bitstream is split into header and side information, motion vectors and DCT encoded signal blocks. Only the latter part of the bitstream is altered; motion vectors and header/side information remain untouched and are copied to the watermarked MPEG-2 bitstream. While other researchers recently proposed to use the motion vectors for watermarking [29], it still has to be proven that such watermarks are persistent after decompression (where the motion information is not present any more). The DCT encoded signal blocks are represented by a sequence of Huffman codes, each representing one (run,level)-pair and, thus, one non-zero DCT coefficient of the current signal block (a special case is the DC coefficient of each block which is encoded differentially to the DC coefficient of the previous block, and with a fixed-length code). Each incoming Huffman code word is decoded (EC^{-1}) and inversely quantized (Q^{-1}), yielding one quantized DCT coefficient of the current signal block. We add the corresponding DCT coefficient from the transformed watermark block, yielding one watermarked DCT coefficient. We then quantize (Q) and Huffman encode (EC) the watermarked coefficient, together with its preceding run of zero coefficients. We compare the number n_1 of bits for the new Huffman codeword with the number of bits n_0 for the old, unwatermarked coefficient. If we do not want to increase the bit-rate of the video bitstream, we only transmit the watermarked coefficient if $n_1 \leq n_0$. Otherwise, we transmit the unwatermarked DCT coefficient and cannot embed the watermark into this DCT coefficient. Since we embed one bit of watermark information into

many pixels and thus many DCT coefficients, we can discard some of them as long as there are enough DCT coefficients of the video left that are watermarked. The algorithm as described and depicted in Figure 9 applies to AC coefficients of intra and inter blocks. DC coefficients of intra blocks are always encoded with a fixed number of bits and are always watermarked. Thus, as a "worst case", the watermark can be embedded into the DC coefficients of intra blocks. If necessary, we can increase the spreading factor, increasing the robustness to the desired level, but at the same time decreasing the data rate for the watermark. We should note that only existing (non-zero) DCT coefficients of the input bitstream are used for watermarking. Among the non-zero coefficients, only those are really watermarked that do not increase the bit-rate. Typically, around $10 - 20\%$ of the DCT coefficients are altered, depending on group-of-pictures structure, bit-rate, and the video sequence. An interesting implication of the fact that only existing (non-zero) DCT coefficients of the input bitstream are watermarked is that the embedded watermark depends on the image signal. In areas where only low spatial frequencies are in the encoded signal, the watermark can contain only low-frequency components, too. This complies with human vision: more watermark signal energy is embedded where it is less visible.

3.2 Drift Compensation

Motion-compensated hybrid coding is a recursive scheme. For intra-coded frames, the mean of each signal block is predicted from previous blocks of the frame. For inter-coded frames, motion compensated predictions from previous frames are used to reconstruct the current frame, which itself may serve as a reference for future predictions, and so on. Once a degradation occurs in the video sequence, it may propagate in time, and even spread in space [30]. Adding a watermark is such a degradation. Even worse, the effects of watermarks from different frames can accumulate. Therefore, a drift compensation signal has to be added besides the watermark signal that compensates for watermark signals from previous frames, as shown in Fig. 10. The signal that has to be added is exactly the difference of the predictions made at coder and decoder. This idea is similarly known in the context of trans-coding of MPEG bitstreams [31]. Figure 11 shows an according extension of the presented watermarking scheme with calculation of the drift compensation signal as the difference of the (motion compensated) predictions from the unwatermarked bitstream (left MC prediction block) and the watermarked bitstream (right MC prediction block). If no watermark is embedded, watermarked and unwatermarked bitstreams are the same, the predictions made from them are the same, and the drift compensation signal is zero. If a watermark is embedded into frame k, yielding frame k', and frame $k+1$ uses frame k' for motion compensated prediction, then the watermark from frame k' can be found in

Fig. 12. Simplified scheme for watermarking of compressed video with drift compensation.

improvement seems possible, if the MC is done in the DCT domain and the forward and inverse cosine transforms in the motion compensation (MC) loop can be omitted. Assunção and Ghanbari have presented such DCT domain MC techniques which offer computational savings up to 81 % [31]. Figure 13 shows the according, fast watermarking scheme incorporating all speed-ups mentioned. Additional complexity savings can be achieved if the watermark signal is directly generated in the DCT domain and inversely transformed by the IDCT only if the watermark has to be retrieved from a decoded version of the video sequence. However, this is not compatible with the schemes as depicted before.

A speed limiting factor of all fast implementations is that the bitstream to be watermarked has at least to be parsed, that means the bitstream has at to be continuously split into its components like header information, motion vector information, and others. We found that parsing of the bitstream alone consumes roughly one third to one half of the execution time of an H.263 or MPEG-2 video decoder. Some operations that a decoder has to perform are not necessary for watermarking, namely reconstruction and display of the decoded images, including IDCT operations. On the other hand, there are additional operations needed for watermarking. Especially DCT of the watermark signal, Huffman decoding and encoding, and the motion compensation loop are the most demanding tasks. Thus, the overall complexity of fast watermarking schemes for compressed video is typically in the same magnitude as for a decoder. As real-time decoders become feasible in low-cost integrated circuits, so does watermarking of compressed video.

Fig. 13. Further simplified scheme for watermarking of compressed video with drift compensation using DCT-domain motion compensation.

3.4 Performance and Robustness

The bit error rate (BER) of the described scheme for compressed-domain embedding can be estimated using the results of section 2.3. However, the equation for the BER (12) has to be modified, taking into account that only a fraction of the watermark can in fact be embedded, due to the additional constraints for compressed-domain embedding. We denote this by a factor ϵ that is an embedding efficiency and gives the relation between average amplitude of the watermark before embedding and average amplitude of the really embedded watermark:

$$\mathrm{BER} = \frac{1}{2}\,\mathrm{erfc}\left(\epsilon \; \frac{\sigma_p \cdot \sqrt{cr} \cdot mean(\alpha_i)}{\sqrt{2} \cdot \sqrt{\sigma_{\bar{v}}^2 + \mu_{\bar{v}}^2}} \right). \tag{14}$$

Thus, $0 \leq \epsilon \leq 1$, where $\epsilon = 1$ means the entire watermark can be embedded in the compressed domain, and $\epsilon = 0$ means nothing can be embedded at all. ϵ depends mainly on the average amplitude of the watermark and the bit-rate of the compressed video, but also on the GOP structure, the sequence, the drift compensation, the allowed excess in bit-rate of the watermarked video sequence compared to the unwatermarked sequence (if any), and implementation details. It does, however, not depend on the chip-rate. Table 3 gives some measured values for ϵ depending on average amplitude $mean(\alpha_i)$ and bit-rate of the compressed ITU-R 601 video sequences. For the simulations, 36 frames of the sequences "Football" and "Flowergarden" were used. No increase of the

		bit-rate of video sequence ($\frac{Mbit}{s}$)							
		2	3	4	5	6	7	8	9
	1	0.0268	0.0275	0.0287	0.0274	0.0297	0.0311	0.0306	0.0319
	2	0.0135	0.0139	0.0143	0.0141	0.0156	0.0170	0.0179	0.0202
watermark	3	0.0090	0.0094	0.0100	0.0106	0.0126	0.0151	0.0175	0.0215
amplitude	4	0.0067	0.0074	0.0082	0.0096	0.0127	0.0164	0.0207	0.0268
	5	0.0055	0.0063	0.0076	0.0101	0.0140	0.0194	0.0251	0.0320
	6	0.0048	0.0057	0.0076	0.0110	0.0160	0.0223	0.0290	0.0365

Table 3
Measured values of ϵ depending on watermark amplitude and bit-rate.

bit-rate through watermarking was allowed. If it is allowed that the bit-rate of the compressed sequence increases during watermarking, the values for ϵ are higher.

An an example, consider the following parameters: if we attempt to embed a watermark into an ITU-R 601 video sequence compressed at $7\frac{Mbit}{s}$, and the watermark has a constant amplitude of 5 (of which only a fraction is embedded), then $\epsilon \approx 0.0194$, according to Table 3. If we further use a binary PN signal with variance 1 for spreading, and the chip-rate is chosen to $633,600$ (such that the watermark data rate R_{WM} is approximately 2 bytes per second), then the compressed domain embedding scheme would embed the watermark with an estimated bit error rate of 5×10^{-3}, according to (14). Measured results confirm the estimated bit error rates according to (14). The results of Table 3 and the example indicate that the watermark data rates for compressed domain embedding are significantly lower than for uncompressed domain embedding and do not exceed a few bytes per second. However, this is sufficient for a lot of envisaged applications [17,18]. Also, the bit error rate of the transmitted bits can be decreased by protecting the watermark bits with an error correcting code.

4 Experimental results

The schemes of Figs. 11 and 12 have been implemented and were used to obtain experimental confirmation of the applicability and robustness of the presented watermarking method.

4.1 Bit Error Rates of Embedded Watermarks

The average measured bit error rates for embedded watermarks confirm the results according to the bit error rate estimates given in (12), for uncompressed domain watermarking, and (14), for compressed domain watermarking. It can, however, be noticed that the bit error rates strongly depend on the sequence contents. For sequences with translatory motion, like the "Flowergarden" sequence, less bit-rate is spent for DCT encoding of the error signal, and, thus, the bit error rate of the watermark is typically higher than for sequences with a lot of non-translatory content changes.

4.2 Characteristics of Embedded Watermarks

Figure 14 gives an impression of the structure of a watermark embedded into a compressed video frame. On top, a MPEG-2 encoded video frame (NTSC resolution, coded at 8 Mbit/s) is displayed. Below, the watermark (amplitude 5, amplified for display) is shown. Third, the actually embedded watermark is shown. Due to the compressed domain constraints, only a fraction of the watermark DCT coefficients could be embedded, which clearly show up as DCT basis functions. The watermark is strongly dependent on the image contents. As was pointed out in section 3.1, and similarly by O'Ruanaidh et al. [9], this complies with human vision. At the bottom, the watermarked frame is displayed.

4.3 Quality of Watermarked Compressed Video - Visual Impression

Figure 15 shows another example frame from a video sequence. On top, the original frame without compression and a detail are displayed. In the middle, the same frame after MPEG-2 encoding and decoding and without an embedded watermark is displayed. At the bottom finally the compressed frame with an embedded watermark is displayed. As can be seen, the watermark results in slightly changed pixel amplitudes which are however not visible except in direct comparison to the unwatermarked image. With appropriate parameters, embedded watermarks are not or only hardly visible.

4.4 Complexity

The complexity of the schemes according to Figs. 11 and 12, as shown in Fig. 16, is much lower than the complexity of decoding plus watermarking in the

pixel domain (which is not considered here) plus re-encoding. For comparison purposes, the complexity of decoding alone is also given. While the absolute computation times (acquired on a SUN SPARCstation 20) are of transient interest, the relation between the computation times, especially between decoding plus re-encoding vs. compressed-domain watermarking, indicate the advantages of compressed-domain watermarking when the material to be watermarked is compressed.

5 Conclusions

In this paper, a scheme for additive spread-spectrum watermarking of video has been presented. Furthermore, a new scheme for watermarking of MPEG-2 compressed video in the bitstream domain has been presented. Working on encoded rather than on unencoded video is important for practical watermarking applications. The basic idea is embedding the watermark in the transform domain as represented in the entropy coded DCT coefficients. We have applied the method as a compatible extension of our watermarking method for uncompressed video, but it can in fact embed any additive signal. Although an existing MPEG-2 bitstream is partly altered, the scheme avoids visible artifacts by addition of a drift compensation signal. The watermark can be retrieved from the decoded sequence and without knowledge of the original. With appropriate parameters, the watermarking scheme in the MPEG-2 bitstream domain can achieve data rates for the watermark of a few bytes/second for ITU-R 601 format video while being robust against friendly or hostile manipulations. The complexity of the scheme is comparable to MPEG decoding. The principle can also be applied to other hybrid coding schemes, such as MPEG-1, MPEG-4, ITU-T H.261, or ITU-T H.263.

References

[1] J. Brassil, S. Low, N. Maxemchuk, and L. O'Gorman. Electronic marking and identification techniques to discourage document copying. In *Proceedings IEEE Infocom '94*, pages pp. 1278 – 1287, 1994.

[2] G. Caronni. Ermitteln unauthorisierter Verteiler von maschinenlesbaren Daten. Technical report, ETH Zürich, Switzerland, August 1993.

[3] G. Caronni. Assuring ownership rights for digital images. In *Proceedings VIS 95, Session "Reliable IT Systems"*. Vieweg, 1995.

[4] L. Boney, A.H. Tewfik, and K.H. Hamdy. Digital watermarks for audio signals. In *Proceedings EUSIPCO 1996*, Trieste, Italy, September 1996.

[5] E. Koch and J. Zhao. Digital copyright labeling: providing evidence of misuse and tracking unauthorized distribution of materials. *OASIS magazine*, December 1995.

[6] E. Koch and J. Zhao. Towards robust and hidden image copyright labeling. In *Proceedings of 1995 IEEE Workshop on Nonlinear Signal and Image processing*, Neos Marmaras, Greece, June 1995.

[7] N. Nikolaidis and I. Pitas. Copyright protection of images using robust digital signatures. In *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing 1996 (ICASSP 96), Atlanta, GA, USA*, May 1996.

[8] I. Cox, J. Kilian, T. Leighton, and T. Shamoon. Secure spread spectrum watermarking for multimedia. Technical Report 95-10, NEC Research Institute, Princeton, NJ, USA, 1995.

[9] J.J.K. Ó Ruanaidh, W.J. Dowling, and F.M. Boland. Watermarking digital images for copyright protection. *IEE Proceedings Vision, Image- and Signal Processing*, 143(4):250–256, August 1996.

[10] M. Kutter, F. Jordan, and F. Bossen. Digital signature of color images using amplitude modulation. In *Proceedings of Electronic Imaging 1997 (EI 97), San Jose, USA*, February 1997.

[11] F. Hartung and B. Girod. Digital watermarking of raw and compressed video. In N. Ohta, editor, *Digital Compression Technologies and Systems for Video Communications*, volume 2952 of *SPIE Proceedings Series*, pages 205–213, October 1996.

[12] F. Hartung and B. Girod. Digital watermarking of MPEG-2 coded video in the bitstream domain. In *Proceedings International Conference on Acoustics, Speech, and Signal Processing (ICASSP 97), Munich, Germany*, April 1997.

[13] G.C. Langelaar, R.L. Lagendijk, and J. Biemond. Real-time labeling methods for MPEG compressed video. In *Proceedings 18th Symposium on Information Theory in the Benelux, Veldhoven, The Netherlands*, May 1997.

[14] R. Ohbuchi, H. Masuda, and M. Aono. Embedding data in three-dimensional polygonal models. In *Proceedings ACM Multimedia '97, Seattle, USA*, November 1997.

[15] H. Berghel and L. O'Gorman. Protecting ownership rights through digital watermarking. *IEEE Computer*, pages 101–103, July 1996.

[16] M. Kobayashi. Digital watermarking: Historical roots. Technical report, IBM Research, Tokyo Research Laboratory, April 1997.

[17] Watermarking for DVD - Call for Proposals. submitted by the Data Hiding Sub-Group of the Copyright Protection Technical Working Group of the DVD consortium, see http:// www.dvcc.com/ dhsg/, July 1997.

[18] MPEG-4 Requirements Group, Call for Proposals for Identification and Protection of Content in MPEG-4. ISO/IEC document JTC1/SC29/WG11 N1714, April 1997.

[19] Paul G. Flikkema. Spread-spectrum techniques for wireless communications. *IEEE Signal Processing*, 14(3):26–36, May 1997.

[20] I. Vattulainen and T. Ala-Nissila. Mission impossible: Find a random pseudorandom number generator. *Computers in Physics*, September 1995.

[21] H.D. Lüke. *Korrelationssignale (in German)*. Springer, 1992.

[22] B. Girod. Psychovisual aspects of image communication. *Signal Processing*, 128(3):239–251, September 1992.

[23] F. Goffin, J.-F. Delaigle, C. De Vleeschouwer, B. Macq, and J.-J. Quisquater. Low-cost perceptive digital picture watermarking method. *SPIE Proceedings 3022: Storage and Retrieval for Image and Video Databases V*, pages 264–277, January 1997.

[24] C. Podilchuk and W. Zeng. Perceptual watermarking of still images. *Proceedings First IEEE Signal Processing Society Workshop on Multimedia Signal Processing, Princeton, New Jersey*, June 1997.

[25] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*. McGraw Hill International editions, 1991.

[26] Q. Wang and R.J. Clarke. Motion estimation and compensation for image sequence coding. *Signal Processing: Image Communication*, 4(2):161–174, Apr. 1992.

[27] B. Friedrichs. *Kanalcodierung (in German)*. Springer, 1996.

[28] T. Sikora. Low complexity shape-adaptive DCT for coding of arbitrarily shaped image segments. *Image Communication, special issue on coding techniques for very low bit-rate video*, 7(4-6), November 1995.

[29] F. Jordan, M. Kutter, and T. Ebrahimi. Proposal of a watermarking technique for hiding/retrieving data in compressed and decompressed video. ISO/IEC document JTC1/SC29/WG11 MPEG97/M2281, July 1997.

[30] B. Girod, N. Färber, and E. Steinbach. *Insights into Mobile Multimedia Communication*, chapter Error-Resilient Coding for H.263. Academic Press, 1997.

[31] P. Assunção and M. Ghanbari. Transcoding of MPEG-2 video in the frequency domain. In *Proceedings International Conference on Acoustics, Speech, and Signal Processing (ICASSP 97), Munich, Germany*, pages 2633–2636, April 1997.

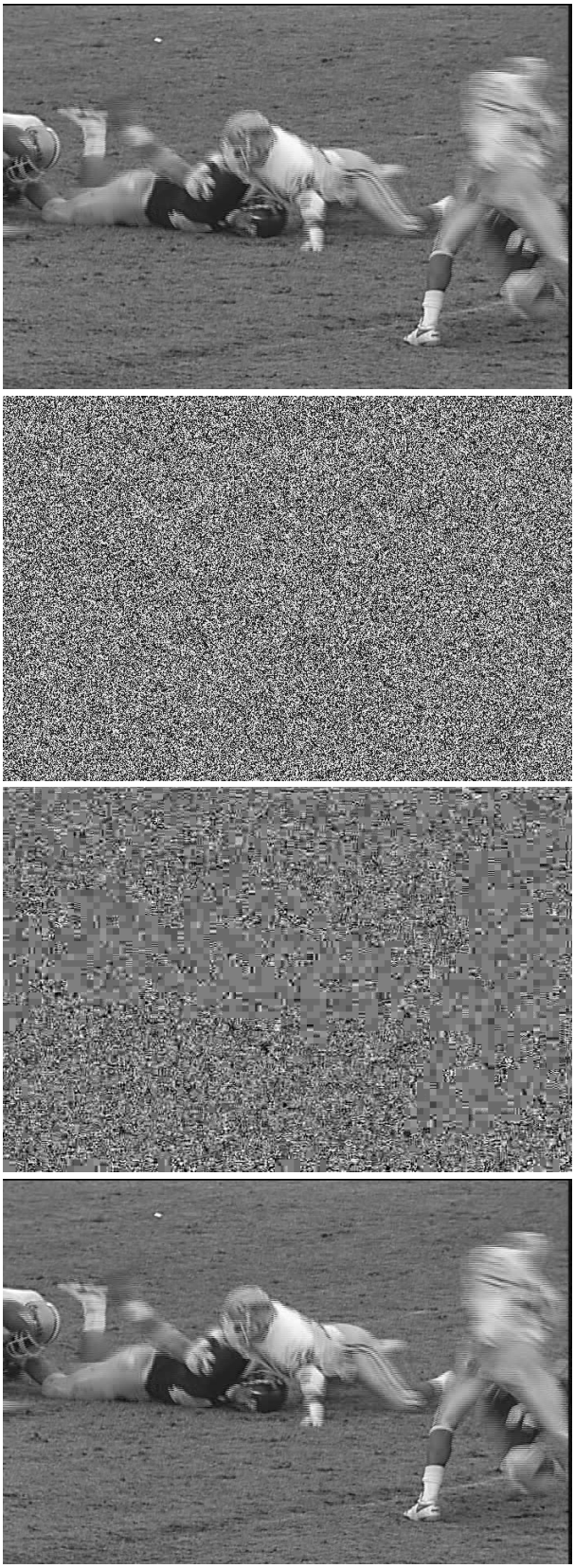

Fig. 14. Example of watermark embedding. Top: MPEG-2 coded frame (NTSC resolution) without watermark, second: watermark generated in the pixel domain (amplified for display), third: actually embedded version of the watermark (amplified for display), bottom: MPEG-2 coded and watermarked frame.

Fig. 15. Example for compressed-domain watermarking. Top: uncoded frame (sequence "table tennis", CIF resolution), middle: MPEG-2 coded frame (2 Mbit/s) without watermark, bottom: the same MPEG-2 coded frame with embedded watermark (watermark amplitude 4).

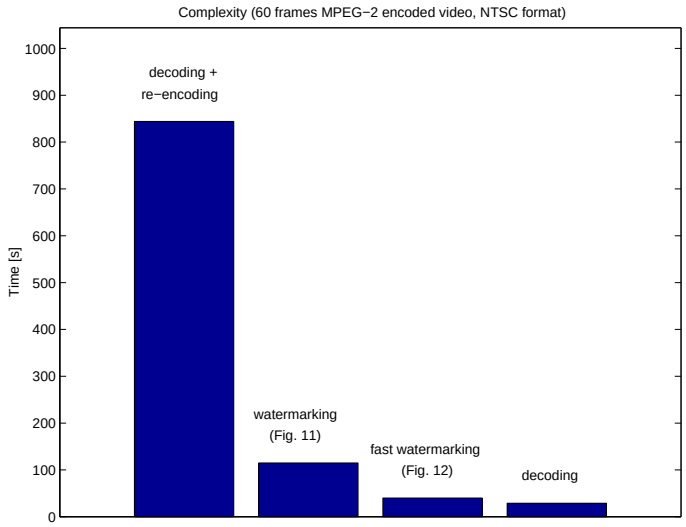

Fig. 16. Complexity of compressed-domain watermarking.