International Journal of Multimedia Data Engineering and Management

October-December 2014, Vol. 5, No. 4

Table of Contents

Research Articles

The *International Journal of Multimedia Data Engineering and Management* is indexed or listed in the following: ACM Digital Library; Bacon's Media Directory; Cabell's Directories; DBLP; Google Scholar; INSPEC; JournalTOCs; Library & Information Science Abstracts (LISA); MediaFinder; The Standard Periodical Directory; Ulrich's Periodicals Directory

Towards Robust Invariant Commutative Watermarking-Encryption Based on Image Histograms

Roland Schmitz, Stuttgart Media University, Stuttgart, Germany

Shujun Li, University of Surrey, Guildford, UK

Christos Grecos, Independent Imaging Consultant, Glasgow, UK

Xinpeng Zhang, Shanghai University, Shanghai, China

ABSTRACT

Invariant Commutative Watermarking-Encryption means to use a cipher that does not have any impact on a certain feature space, which can thus be used for embedding watermarks either before or after encryption. For example, histogram-based watermarking schemes are invariant to pixel permutations and can be combined with permutation-based ciphers to form a Commutative Watermarking-Encryption (CWE) scheme. However, typical histogram-based watermarking schemes based on comparison of histogram bins are prone to de-synchronization attacks, where the whole histogram is shifted by a certain amount. In this paper the authors investigate the possibility to avoid this kind of attacks by synchronizing the embedding and detection processes, using the mean of the histogram as a calibration point. The resulting watermarking scheme is resistant to three common types of shifts of the histogram, while the advantages of previous histogram-based schemes, especially commutativity of watermarking and permutation-based encryption, are preserved. The authors also report on the results of testing robustness of the scheme against JPEG and JPEG2000 compression.

Keywords: *Commutative Watermarking-Encryption (CWE), De-Synchronization Attacks, Histogram-Based Watermarking, Image Histogram, Permutation-Based Encryption*

1. INTRODUCTION

Encryption and watermarking are both important tools in protecting digital contents, e.g. in digital rights management (DRM) systems. While encryption is used to protect the contents from unauthorized access, watermarking can be deployed for various purposes, ranging from ensuring authenticity of content to embedding metadata, e.g. copyright or authorship information, into the contents. Heterogeneous end-to-end media distribution scenarios, where

DOI: 10.4018/ijmdem.2014100103

the ultimate receiver of the media data may be unknown to the sender, call for protection schemes in which both watermarking and encryption need to be combined in a flexible way.

The concept of commutative watermarking-encryption (CWE) was first discussed in (Herrera-Joancomarti et al., 2005) with a special emphasis on watermarking in the encrypted domain. Four properties about watermarking in the encrypted domain are formulated in Sec. 2.2 of Herrera-Joancomarti et al.'s report:

Property 1: The marking function M can be performed in the encrypted domain;
Property 2: The verification function V is able to reconstruct a mark in the encrypted domain when it has been embedded in the encrypted domain;
Property 3: The verification function V is able to reconstruct a mark in the encrypted domain when it has been embedded in the clear domain;
Property 4: The decryption function does not affect the integrity of the watermark.

All four properties should hold without the marking and verification functions having access to the encryption key, and without the encryption and decryption functions having access to the watermarking key. The four properties can be fulfilled in the most natural way if the encryption operation and the watermarking operation commute, meaning that the outcome is the same no matter whether the encrypted media are watermarked or if the watermarked media are encrypted (see also Sec. 4.4).

The present paper is mainly concerned with a histogram-based watermarking scheme that is capable of being integrated into a CWE scheme. It is well known that histogram-based watermarking schemes are resistant to permutations of image pixels. In particular, using histograms implies robustness against rotation, scaling and translation (RST) of images. In (Schmitz, 2012) this fact has been utilized to devise a commutative watermarking-encryption (CWE) scheme by choosing a permutation cipher for encryption and a histogram-based scheme for watermarking.

However, typical histogram-based watermarking schemes like those proposed in (Schmitz et al., 2012) and (Chrysochos et al., 2007) work by comparing selected histogram bins, where the selection process is controlled by a watermarking key. If the whole histogram is shifted by a small amount, i.e. by adding a small number to each pixel value, the detector will use completely different bin pairs for extracting the embedded watermark and will produce wrong results. To overcome this problem, in the present paper, we further improve our earlier work described in (Schmitz et al., 2012) by deploying a synchronization process between embedder and detector that is based on the global mean of the histogram. A preliminary edition of the present paper was presented at the 2013 IEEE International Symposium on Multimedia (ISM), see (Schmitz et al., 2013).

The rest of the paper is organized as follows. In Sec. 2 we briefly summarize previous approaches to CWE along with other histogram-based watermarking algorithms. Section 3 describes the three types of histogram shifts we have investigated, and Sec. 4 describes the proposed algorithm in greater detail. In Sec. 5 we discuss experimental results for the algorithm, especially its robustness against histogram shifts and lossy compression. In Sec. 6 we provide a qualitative comparison between the watermarking scheme proposed in the present paper and previous histogram-based schemes. Section 7 concludes the paper and gives directions for further work.

2. RELATED WORK

2.1. Commutative Watermarking-Encryption (CWE)

While encryption algorithms are evaluated mainly according to their security and run-time performance, in watermarking there are more and often conflicting requirements (Cox et al., 2007): *Watermarking security* normally refers

to the difficulty for an attacker to remove/manipulate a watermark that he does not have access to or to insert a watermark of his own, while *robustness* is the ability of a watermark to withstand compression or other image-procession operations. Further evaluation criteria for watermarks are *fidelity* (the degree of imperceptibility) and *capacity* (the number of bits that may be embedded by a watermarking scheme).

One possible way to combine watermarking and encryption is provided by deploying so called homomorphic encryption techniques (Fontaine & Galand, 2007). They are characterized by the property that some basic algebraic operations such as addition and multiplication on the plaintexts can be transferred onto the corresponding ciphertexts, i.e., they are transparent to encryption (cf. (Lagendijk et al., 2013) and Sec. 2.1 of (Herrera-Joancomarti et al., 2005)). Now, if both the encryption and the watermarking processes use compatible homomorphic operations, one gets a commutative watermarking-encryption scheme. Examples of compatible homomorphic operations are exponentiation modulo n (being compatible with multiplication modulo n), multiplication modulo n (being compatible with multiplication modulo n) and addition modulo n (being compatible with addition modulo n), which includes the bitwise XOR operation. One drawback of using addition modulo n for encryption and watermarking is that the modular addition operation may cause overflow/underflow pixels that have to be handled separately, thus making the system "quasi-commutative" (see (Lian, 2009)). The XOR operation does not suffer from the overflow/underflow problem, though. On the other hand, the watermark's robustness is limited by the encryption operation.

In partial encryption based CWE schemes, the plaintext multimedia data is partitioned into two disjoint parts, where one part is encrypted and the other part is watermarked. Since the encryption part is independent of the watermarking part, they are naturally commutative. To take a typical example, in (Lian et al., 2006), the multimedia data is partitioned into two parts after a four-level discrete wavelet transformation. The lowest-level coefficients are fully encrypted, while for the medium- and high-level coefficients only the signs are encrypted. In this case, the unencrypted absolute values of medium-level coefficients can be watermarked either before or after encryption.

Another more recent example of a partial encryption based CWE scheme is the CEWoD (CWE based on Orthogonal Decomposition) scheme (Xu et al., 2014). In this scheme, the media data are organized as an n-dimensional vector X. If B is an orthogonal transformation matrix, X can be written in the form $X = B \cdot Y$, where $Y = B^{-1} \cdot X$. By choosing sub-matrices R and S so that $B = (R, S)$, X gets decomposed into two independent parts:

$$X = B \cdot Y = R \cdot Y_1 + S \cdot Y_2$$

Encryption and watermarking are applied to the orthogonal decomposition coefficients Y_1 and Y_2, respectively.

In (Boho et al., 2013) a partial encryption scheme is used to protect an H.264/AVC & HEVC video stream: Sign bits of DCT coefficients, motion vector differences and prediction modes of I-, B- and P-frames are encrypted, while the residual DCT coefficients are watermarked by Quantized Index Modulation (QIM, see (Chen & Wornell, 2001)).

Generally, in a partial encryption based CWE scheme, there is a trade-off between security of the encryption part and robustness of the watermarking part. The more visually important data are encrypted, the less quality distortion is introduced in the process of watermark removal by an attacker.

The third approach to CWE as introduced in (Schmitz et al., 2012) is to encrypt all media data, but to use a cipher that leaves a feature space invariant. This feature space can be used to embed a watermark. This approach is called invariant encryption in (Boho et al., 2013). As an example, a permutation cipher is used for encryption and a histogram-based algorithm is used for watermarking in (Schmitz et al.,

2012). The advantage of the invariant encryption approach is that all media data are encrypted (and not just a subset); however, by suitable operations in the invariant feature space the watermark may still be removed or manipulated.

2.2. Histogram-Based Watermarking

As in this contribution the watermarking algorithm presented in (Schmitz et al., 2012) will be developed further towards higher robustness, we also give a review of common histogram-based watermarking algorithms.

The most widely studied approach to histogram-based watermarking is so-called exact histogram specification (Coltuc & Bolon, 1999; Roy & Chang, 2004; Lin et al., 2006), where the histogram of the original image or a (randomly and secretly selected) sub-region of it is modified toward a target histogram, which is then used as the signature for watermark detection. However, exact histogram specification does not involve a secret embedding/detection key, and there are few other histogram-based watermarking algorithms without this problem.

For example, the video watermarking scheme by (Chen et al., 2009) first computes the mean value of luminance values for each frame in the video sequence, then computes a histogram for the temporal sequence of the mean values and embeds the watermark by comparing and modifying neighbouring histogram bins. A watermark key is not used for embedding.

The scheme proposed by (Xiang et al., 2008) represents the histogram shape as the ratios of population between groups of two neighbouring bins and then modifies the ratios to carry a key-based pseudo-random sequence. Only histogram bins in the range $\left[(1-\lambda)\bar{A},(1+\lambda)\bar{A}\right]$ are used in the process, where $\bar{A}$ is the global mean of the histogram and $\lambda \in [0.5, 0.7]$ is a public parameter (note that λ must be a multiple of $1/\bar{A}$). In order to withstand scaling and cropping attacks on the image that will also affect the histogram and the mean, the extraction algorithm uses a search process based on the mean $\bar{A}'$ of the histogram of the marked image. Different ranges:

$$[(1-\lambda)(\bar{A}'+s),(1+\lambda)(\bar{A}'+s)]$$

where s is an integer running through some search space, are tried, until the correlation between the extracted sequence and the known embedded sequence reaches a maximum. The resulting scheme is very robust against geometric image modifications and lossy compression, but it suffers from two severe limitations: The parameter λ may be seen as a watermarking key if kept secret, but as λ is taken from the interval $[0.5, 0.7]$, this interval can searched through for candidate values of λ using a stepsize of $1/\bar{A} \approx 1/128$. Thus, there are only $\left(0.7-0.5\right)\times 128 \approx 26$ possibilities for λ, and the correct value of λ can be verified by an attacker by matching the detected watermark to the embedded watermark, if the latter is known. The effective capacity of the scheme is only 20-30 bits. Moreover, as this scheme calculates the image histogram only after filtering out high-frequency information using a Gaussian kernel low-pass filter (and writing this information back into the image after watermarking), it cannot be used in a CWE scheme because encrypted and unencrypted images will contain different high-frequency information.

Two other histogram-based watermarking schemes do use a longer watermarking key, but are by construction prone to histogram shift attacks: The scheme proposed by (Chrysochos et al., 2007) is based on the idea of (selectively) swapping two selected histogram bins *a* and *b* containing the pixels with values *a* and *b*, respectively. In this scheme, the distance $|b-a|$ between *a* and *b* is a fixed number $d < 10$. A message bit is encoded by the relative heights of the bins (denoted by $hist(a)$ and $hist(b)$: a 1-bit is encoded by $hist(a) > hist(b)$ and a 0-bit by $hist(a) < hist(b)$. Here, swapping two histo-

gram bins *a* and *b* means changing all pixel values *a* to *b* and vice versa.

In (Schmitz et al., 2012), the scheme described in (Chrysochos et al., 2007) was extended and integrated into an invariant encryption based CWE scheme. Histogram bins *a* and *b* are randomly selected from the 256 available bins under the condition that their relative distance *d* is smaller than 10. This leads to a significant enlargement of the key space. As this scheme also forms the basis for the present watermarking algorithm it is described in greater detail in Sec. 4.

3. HISTOGRAM SHIFT ATTACKS

Most histogram-based watermarking algorithms work by analyzing the image histogram at a certain location defined by a watermarking key. If an attacker manages to de-synchronize embedder and detector such that they analyze different parts of the histogram, the watermark cannot be detected anymore. In this section we describe simple histogram modification attacks, where the histogram as a whole is shifted along the horizontal axis by adding a (positive or negative) fixed amount to each pixel value. As mentioned above, in principle these attacks are relevant for all histogram-based watermarking algorithms that rely on manipulating certain pre-defined histogram bins. We demonstrate this fact for the earlier schemes by (Chrysochos et al., 2007) and (Schmitz et al., 2012) in Sec. 5.

We differentiate among three ways of histogram shifting, where we focus the discussion on gray-scale images with a bit-depth of 256 for the sake of simplicity. The basic principle can be easily generalized to color images and images with a higher bit depth.

3.1. Cyclic Histogram Shifting

In a cyclic shift, each pixel value *P(i,j)* is shifted by a certain amount *x* modulo 256:

$$P_{\text{attacked}}(i,j) = (P(i,j) + x) \mod 256$$

where *x* is a positive or negative integer. Due to the wrap-up at both edges of the histogram, cyclic histogram shifting may lead to severe degradation of image quality (see Figure 1(b)). Cyclic histogram shifts therefore constitute practically less relevant attacks. Moreover, cyclic shifts are invertible if the amount of shift is known. This fact will be used later, when we try to approximate non-cyclic shifts with suitable cyclic shifts.

3.2. Accumulated Non-Cyclic Histogram Shifting

A more relevant attack leading to less degradation of image quality is accumulated non-cyclic histogram shifting. Here, the wrap-up in cyclic shifting is avoided as the shift is only applied to those pixels whose gray values are sufficiently small or large. For example, a rightward shift can be defined by:

$$P_{\text{attacked}}(i,j) = \begin{cases} P(i,j) + x, & \text{if } P(i,j) < 256 - x \\ P(i,j) & \text{else} \end{cases}$$

where *x* is a positive integer chosen by an attacker. Analogously, a leftward shift may be defined by using a negative integer *x*. This kind of histogram modification leads to an accumulation of pixels at the start or the end of the histogram. Nevertheless, the amount of image distortion remains small, if $| x |$ is sufficiently small. Note that this kind of histogram shift normally cannot be fully reverted, unless sufficient information about the original histogram is known.

3.3. Histogram Cropping

In a histogram cropping attack, the bins are shifted to the left (or to the right), where bins are dropped if their corresponding pixel value exceeds 255 or falls below zero. To be specific, a rightward shift-and-crop operation can be defined by the new histogram:

$$H_{\text{attacked}}(i) = \begin{cases} 0, \text{ if } 0 \le i \le x-1 \\ H(i-x), \text{ else} \end{cases}$$

The parameter x can be seen as a measure for the amount of shift, before the cropping takes place. The resulting histogram for small $|\,x\,|$ is similar to the original one (see Figure 1(d)), but contains less pixels and therefore does not constitute a valid histogram for the original image. However, after rescaling the attacked histogram, the attacked image can be reconstructed from the attacked histogram by exact histogram specification (Coltuc & Bolon, 1999).

3.4. Comparing the Shifts

Figure 1 shows the visual influence of the three kinds of histogram shift on an example image. In all cases, the blue channel histogram has been shifted by an amount of 20. Note the visible artifacts in the case of a cyclical shift, while the other two shifts do not have any visible effect.

Despite the different visual influence, the three kinds of shifts can behave very similarly up to a certain amount of shift, depending on the histogram shape. Figure 2 shows the effects of the various kinds of shifts on the same example histogram (the blue channel histogram of the baboon image, see Figure 2(a)).

We also computed the quantitative amount of distortion caused by each type of histogram shift by calculating the PSNR (Peak Signal-to-Noise-Ratio) between the blue channel of the original image P and the blue channel of the attacked image P_{attacked} computed from the shifted blue channel histogram. The PSNR is for monochromatic images consisting of $(m \times n)$ pixels defined by:

$$PSNR = 10 \log_{10} \left(\frac{\left(\max_{1 \le i \le m, 1 \le j \le n} P(i,j) \right)^2}{MSE} \right),$$

$$\text{where } MSE = \frac{1}{m \cdot n} \sum_{i=1}^{m} \sum_{j=1}^{n} \left[P(i,j) - P_{\text{attacked}}(i,j) \right]^2$$

is called the Mean Squared Error. In Figure 3 the resulting PSNR is plotted against the amount of shift for the three types of shift and for two different test images. Figure 3(b) shows a very similar behaviour of all three shift types, while in Figure 3(a) there is a marked difference between the graph of the cyclic shift (shown in red) and the other two types. Shifting the red graph horizontally, however, will bring it very close to the other two graphs, meaning that the non-cyclic shifts and histogram crops can be approximated by cyclic shifts of a slightly different amount.

As cyclic shifts can be reverted, it seems reasonable to assume that the effects of non-cyclic shifts and histogram crops on the water-

Figure 1. Effects of a histogram shift by the amount of 20: (a) Original image; (b) Cyclical shift (PSNR: 34.22); (c) Non-cyclical shift (PSNR: 41.09); (d) Shifted and cropped blue channel histogram (PSNR: 42.46)

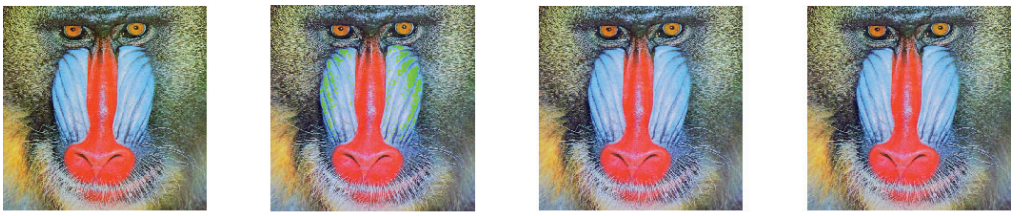


Figure 2. Effects of a histogram shift by the amount of 20: (a) Original histogram; (b) Cyclically shifted histogram; (c) Non-cyclically shifted histogram; (d) Shifted and cropped histogram

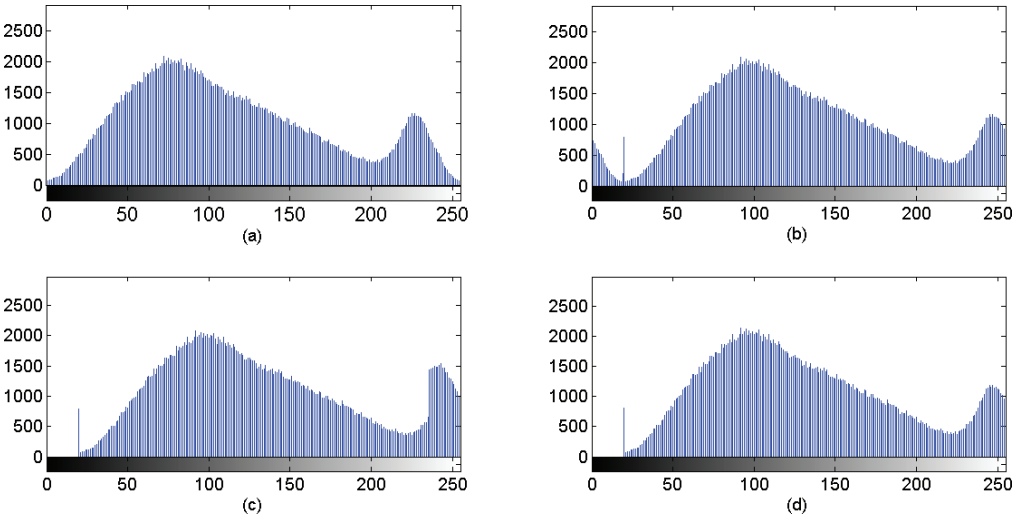
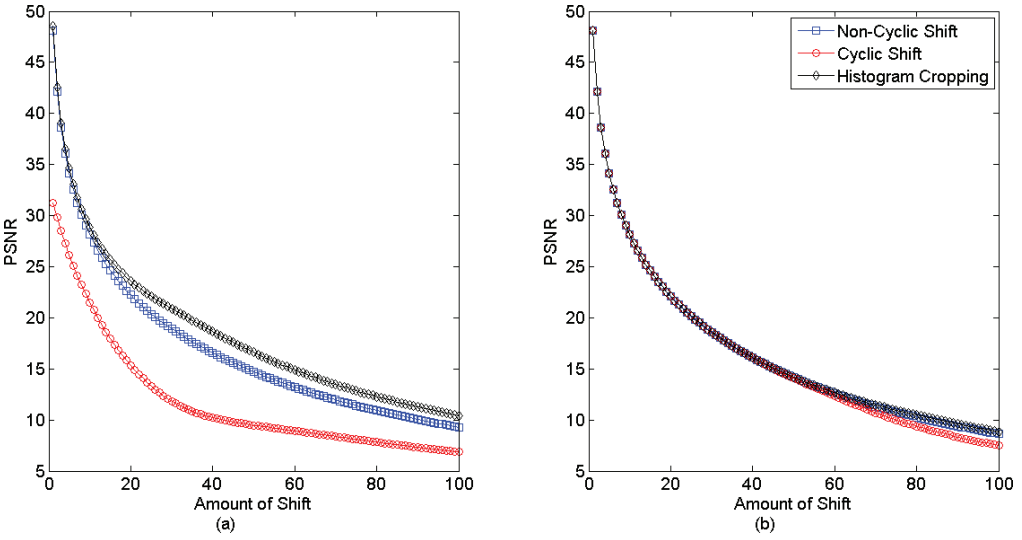


Figure 3. The amounts of distortion caused by histogram shifts: (a) Blue channel of Baboon image; (b) Blue channel of Lenna image



marked image can be approximately reverted by a suitably chosen cyclic histogram shift at the detector side. The optimal cyclic shift amount is found when the linear correlation of the detected mark and the reference mark reaches the maximum (see Sec. 4).

4. PROPOSED CWE ALGORITHM

The design goal of the presented algorithm is to improve the robustness of previous histogram-oriented CWE algorithms against

simple histogram shifts, while retaining the original advantages, especially commutativity of watermarking and permutation-based encryption. Moreover, the algorithm should be able to use a watermarking key that is long enough to withstand brute-force attacks.

4.1. Embedding

In order to embed a given bipolar *N*-bit watermark $W = \{w_i\}, 1 \leq i \leq N$ (i.e. $w_i \in \{-1,1\}$), a single watermark bit w_i is embedded by pseudo-randomly selecting two histogram bins that have not been selected before, if their distance is smaller than some parameter *d* and if they are not of equal height. Here, bipolar watermarking bits were chosen in order to facilitate watermark detection via linear correlation. The heights of the two selected bins a_i and b_i encode w_i as follows: if w_i = 1, hist(a_i) < hist(b_i) should hold, and if w_i = -1, hist(a_i) > hist(b_i) should hold, where hist(*x*) denotes the height of bin *x*. If this is not the case, the two bins a_i and b_i are swapped. The selection process for the bin pairs is governed by a watermark key W_K. The theoretical maximum capacity that can be achieved by this scheme is 128 bits and can be further extended by using more than one color channel and/or subdividing the image.

In order to speed up the search for the optimal amount of cyclic shift during extraction, we devised a calibration process that uses the global mean value $\overline{A}$ of the image as a calibration point. More specifically, before selecting the histogram bins for embedding, all bins are cyclically shifted by an amount of $x = 256 - \overline{A}$ as described in Sec. 3 so that the bin corresponding to $\overline{A}$ becomes the first bin in the calibrated histogram. After calibrating, the embedding process proceeds as described above.

4.2. Detection

Basically, the detector works by comparing the histogram bins as specified by the watermarking key. For this to work the embedder and the detector need to be synchronized, i.e. they need to use the same ordering of histogram bins. As a histogram shift will change the global mean gray value of the watermarked image, the detector searches for the correct calibration point by cyclically shifting the histogram of the watermarked image by an amount of $x = 256 - \overline{A'} - s$, where $\overline{A'}$ is the mean value of the marked shifted image, and *s* runs through the search space:

$$S = \{s \mid -\overline{A}' / 4 \leq s \leq \overline{A}' / 4\}$$

If needed, this search space may be enlarged until it covers the complete range $0 \leq s \leq 255$ (see also Sec. 5). The detector then computes the linear correlation of the extracted watermark $W_{\text{ex}} = \{\tilde{w}_i\}$ with the reference watermark $W = \{w_i\}$ for each *s*. The detector response is:

$$\max_{-\bar{A}'/4 \leq s \leq \bar{A}'/4} \left(\frac{1}{N} \sum_{i=1}^{N} \tilde{w}_i w_i \right)$$

As usual in correlation based detection, the watermark is detected if the detector response exceeds a certain previously chosen threshold *T*. A larger *T* means a lower false positive probability and a higher false negative probability (see below). While the synchronization process proposed by (Xiang et al., 2008) is very similar to the calibration process described here, the present approach has a much larger key space and a higher capacity.

4.3. Keyspace and False Positive Probability

As the calibration process prior to selecting the histogram bin pairs does not affect the number of available bin pairs, the key space size *K(N)*, where *N* is the length of the watermark, stays the same as in the earlier scheme (Schmitz et al., 2012). In order to get a lower bound on *K(N)*, we partition the 256 histogram bin into *N* disjoint parts, each containing $256/N$ bins.

In order to embed a single bit, we choose a bin pair from each part. The members of the bin

pair must not be further than d apart. As there may be less than d bins in each part, there are:

$$\frac{256}{N} \times \min\left\{d, \frac{256}{N} - 1\right\}$$

possibilities to define a bin pair. Repeating this for each watermarking bit and noticing that the N bits may be arbitrarily permuted, gives an upper bound of:

$$K\left(N\right) > N! \left(\frac{256}{N} \times \min\left\{d, \frac{256}{N} - 1\right\}\right)^N$$

For $N = 32$ and $d = 10$, the key length is already well beyond 200 bits.

In order to simplify the analysis of the false positive probability, we assume that the bipolar bits of W_{ex} are evenly distributed in the set $\{-1,1\}$. Then, the probability that two single bits of W and W_{ex} agree is 1/2. Now let $\tilde{W}$ be a mark extracted from an unmarked image I_U. If $\tilde{W}$ agrees with W at k positions, their linear correlation is $\frac{2k - N}{N}$. Therefore, the mark is wrongly detected if $k > \frac{N}{2}\left(T + 1\right)$. Thus, the false positive probability for a single detection step becomes:

$$p(\text{FalsePos}) = q = \left(\frac{1}{2}\right)^N \cdot \sum_{k=\left\lceil \frac{N}{2}(T+1) \right\rceil}^{N} \binom{N}{k}$$

For example, if $N = 64$, choosing the detection threshold $T = 0.7$ and evaluating the formula above leads to a false positive probability $q \leq 1.77 \times 10^{-9}$, while choosing a lower bound $T = 0.3$ gives $q \leq 0.0084$. If the detection process is carried out by running through a search space of size $|S|$, the false positive probability becomes:

$$p(\text{FalsePos}) = 1 - (1 - q)^{|S|} \approx q\,|S| \quad \text{for small } q$$

On the other hand, false negative probabilities in general are highly dependent on what happens to the watermarked image between the time the watermark was embedded and the time it is detected. The watermark can be distorted by lossy compression and other image processing operations, resulting in an increased probability of a false negative (cf. (Cox et al., 2007)). It is therefore very difficult to give an analytical estimation of the false negative probability in general. Instead, we will experimentally assess the robustness of the proposed watermarking algorithm against histogram shifts and lossy compression in Sec. 5. A higher robustness of the watermark automatically results in a lower false negative probability.

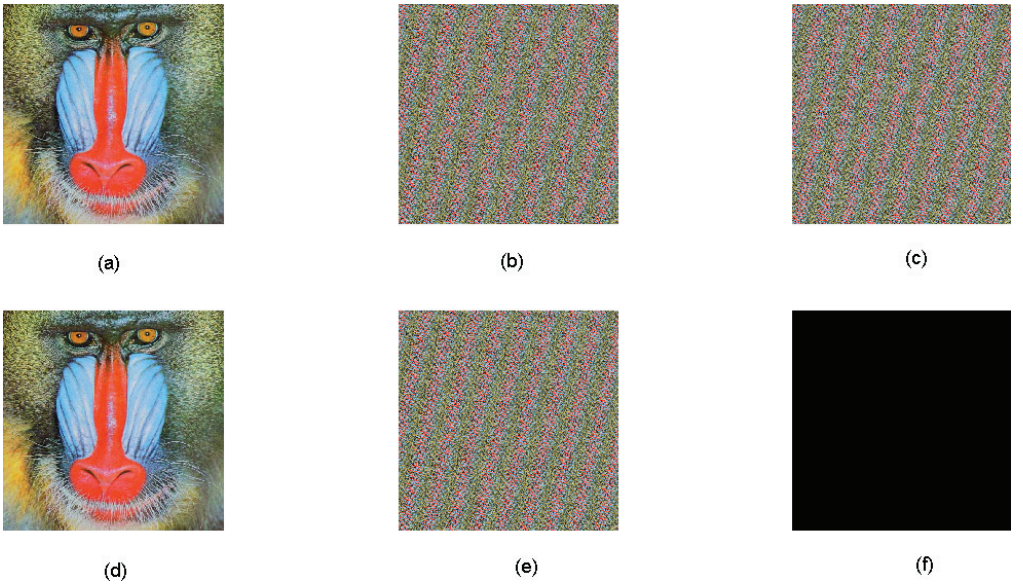
4.4. Commutativity with Permutation-Based Encryption

As the presented algorithm is completely histogram-based and the histogram is invariant to permutations, the commutativity property:

$$M(E_k(I), m) = E_k(M(I, m))$$

holds, where E is the encryption function, k is the encryption key, I is the plaintext media data, M is the watermarking function and m is the mark to be embedded. Figure 4 shows an example encryption-watermarking process. In the upper row, the plaintext image I is first encrypted and watermarked afterwards. The result (Figure 4(c)) is $M(E_k(I), m)$. In the lower row, on the other hand, we watermark the plaintext image first and encrypt it afterwards, resulting in $E_k(M(I, m))$ (see Figure 4(e)). Finally, Figure 4(f) shows the difference image of

Figure 4. Commutative watermarking-encryption process for the baboon image: (a) Plaintext image; (b) Encrypted image; (c) Watermarked encrypted image; (d) Watermarked plaintext image; (e) Encrypted watermarked image; (f) Difference image of (c) and (e)



$M(E_k(I), m)$ and $E_k(M(I, m))$ where all pixels have colour values zero.

In our prototype implementation, we used a permutation generated by the repeated application of a discrete two-dimensional chaotic map, the so-called Cat Map (see (Schmitz et al., 2012) and (Chen et al., 2004) for details) for encryption.

5. EXPERIMENTAL RESULTS

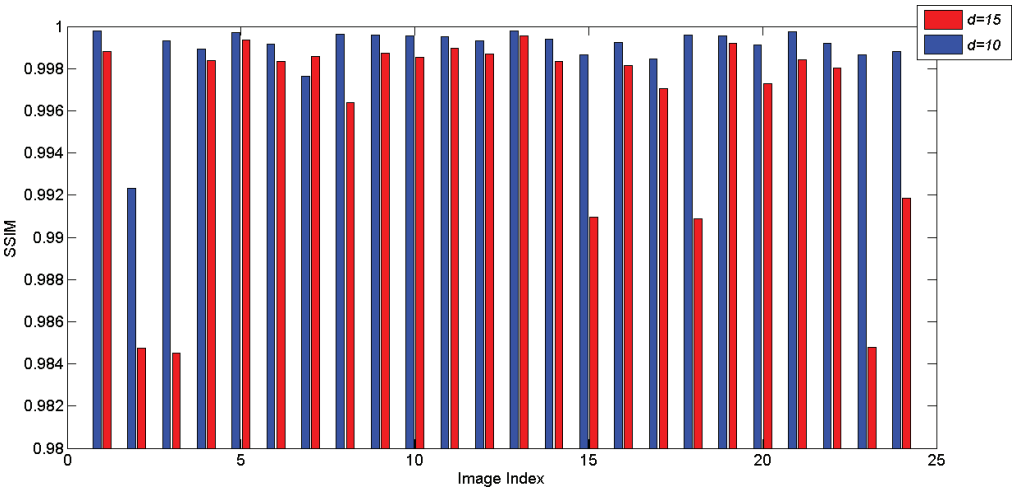
In our experiments, we embedded 64 random bits into the blue channel of all the 24 images from the Kodak image database (see http://r0k.us/graphics/kodak) and three additional standard images from the SIPI image database (a collection of digitized images hosted at the Signal and Image Processing Institute of the University of Southern California, see http://sipi.usc.edu/database/). We measured the amount of visual distortion for two different maximum step sizes *d* and investigated robustness against histogram shifts and JPEG/JPEG2000 compression.

5.1. Visual Distortion

The main difference between the present algorithm and the algorithm proposed in (Schmitz, 2012) consists in the calibration step performed before embedding and detecting, which has no impact on the amount of distortion. Therefore, results on visual distortion basically carry over from (Schmitz et al., 2012).

In order to assess the overall visual quality, we chose Structural Similarity Index (SSIM, cf. (Wang et al., 2004)) as the metric, because SSIM correlates better to the human visual system than the traditionally more used Peak Signal-to-Noise Ratio (PSNR). Figure 5 shows the SSIM values for the 24 images from the Kodak database after embedding 64 random bits, where embedding was carried out with two different maximum step sizes $d = 10$ and $d = 15$, respectively. Clearly, visual distortions

Figure 5. Visual quality comparison of embedding 64 bits into 24 test images with maximum stepsizes $d = 10$ (blue bars) and $d = 15$ (red bars)



are higher for the greater stepsize, but they remain low for both stepsizes.

Figure 6 shows the visual effect of embedding 64 random bits into the blue channel of three standard images from the SIPI database with a maximum stepsize $d = 10$. Only the marked images are shown.

5.2. Robustness against Histogram Shifts

First, we tested robustness of two earlier algorithms (Chrysochos, 2007) and (Schmitz, 2012) against histogram shifts. As expected, the watermark is completely destroyed in most cases, resulting in a correlation value around or even below zero (see Figure 7).

When testing the improved algorithm described in Sec. 4, it turned out that the three standard images from the SIPI database (Baboon, Sailboat and Lenna) show a rather prototypical behaviour with respect to robustness against the three kinds of histogram shift attacks. Figures 8(a) and 8(b) show a very good robustness for the sailboat and Lenna image, due to the fact that the corresponding histograms behave very

Figure 6. Embedding 64 random bits into three standard test images: (a) Watermarked baboon image (PSNR: 50.55 dB); (b) Watermarked sailboat image (PSNR: 58.86 dB); (c) Watermarked Lenna image (PSNR: 58.60 dB)

Figure 7. Robustness of previous histogram-based watermarking algorithms against histogram shifts: (a) Algorithm by Chrysochos et al. (2007); (b) Algorithm by Schmitz et al. (2012)

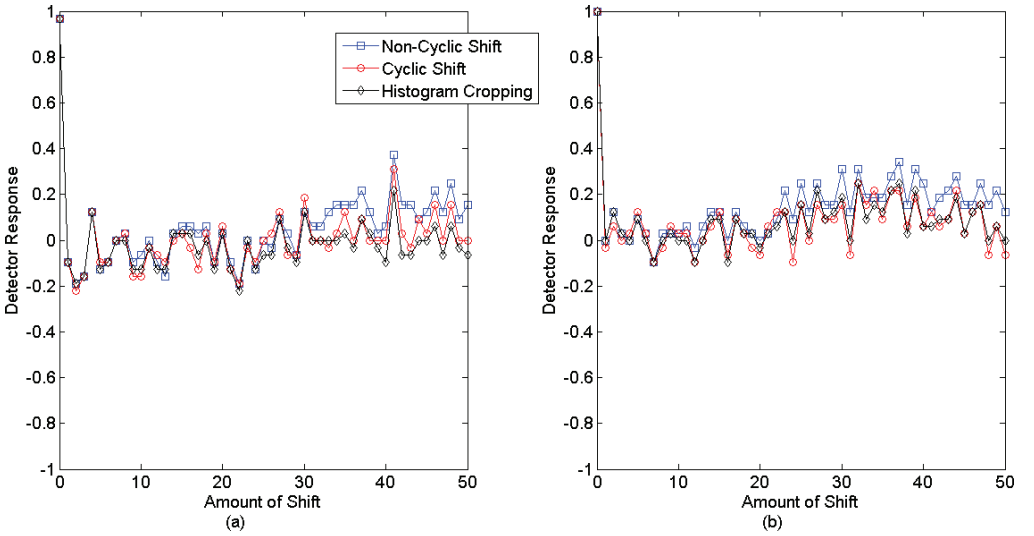
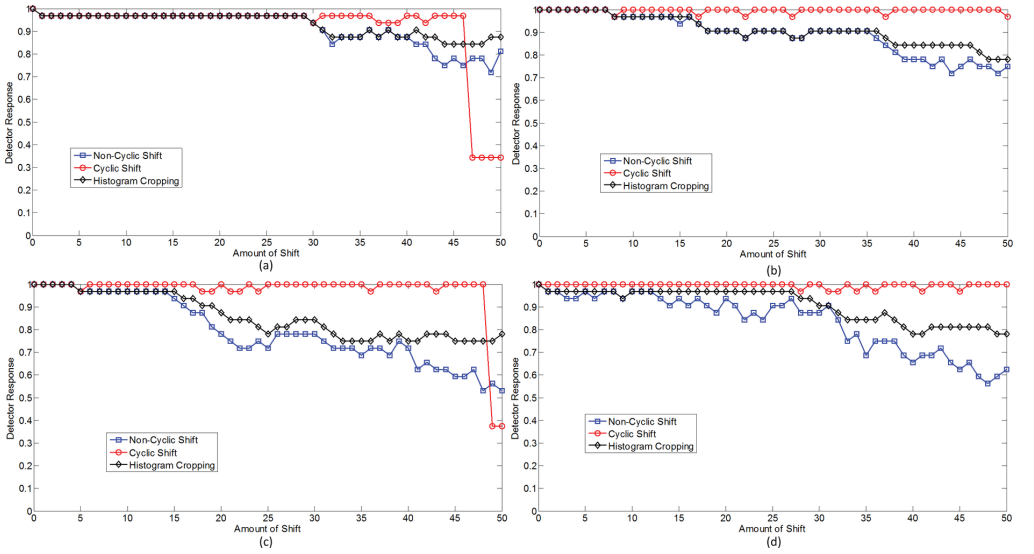


Figure 8. Robustness results against histogram shifts: (a) Sailboat image; (b) Lenna image; (c) Baboon image; (d) Baboon image with an enlarged search space



similarly for the three kinds of shift and the shifts may thus be reversed by a suitable cyclic shift quite accurately. The relevant histogram for Figures 8(c) and 8(d) is the blue channel histogram of the baboon image (see Figure 2(a)), which behaves less favourably. In this case, the results can be significantly improved by enlarging the search space, e.g. to:

$$S = \{s \mid -\bar{A}' / 2 \leq s \leq \bar{A}' / 2\}$$

as Figure 8(d) shows. All other tested images behaved in a similar way. Thus, a robust detection strategy consists in enlarging the search space successively if the mark is not detected using a smaller search space. The resulting increase of the false positive probability is negligible (see Sec. 4).

In order to get an overall impression of the robustness against histogram shifts, Figure 9 gives the average results of the tests with the 24 images in the Kodak database. Even with a shift amount of 50, the watermark can still be reliably detected with a threshold value of $T = 0.7$.

5.3. Robustness against Compression

In this part of the section we report the robustness of the proposed watermarking algorithm against JPEG and JPEG2000 compression. As Figure 10 shows, the histograms are affected even by mild compression in a quite serious way that is similar to adding random noise to the image. Different from histogram shifts, there is no connection to cyclic shifts and the calibration process described above will not have an effect. Therefore, it can be expected that the robustness against compression of watermarking algorithms based on comparing histogram bins is not very high.

Figure 11 shows the detector results when an uncompressed image (the Lenna image) is watermarked and afterwards compressed using JPEG and JPEG 2000 with varying quality degrees. Watermarking was done with the previous algorithms (Chrysochos, 2007) and (Schmitz, 2012). While the later algorithm gives better detector responses, it is clear from these results that neither algorithm is robust with respect to JPEG / JPEG2000 compression.

Figure 12, on the other hand, shows the corresponding results for the algorithm proposed in the present paper. Here, watermarking was done with two different stepsizes to see if increasing the step size can improve robustness against lossy compression.

As the results in Figure 12 show, there is some improvement compared to the earlier algorithms (Chrysochos, 2007) and (Schmitz, 2012), leading to a modest degree of robustness against JPEG/JPEG2000 compression. The robustness against JPEG2000 compression can be slightly increased if a bigger stepsize for

Figure 9. Robustness results against histogram shifts averaged over 24 test images

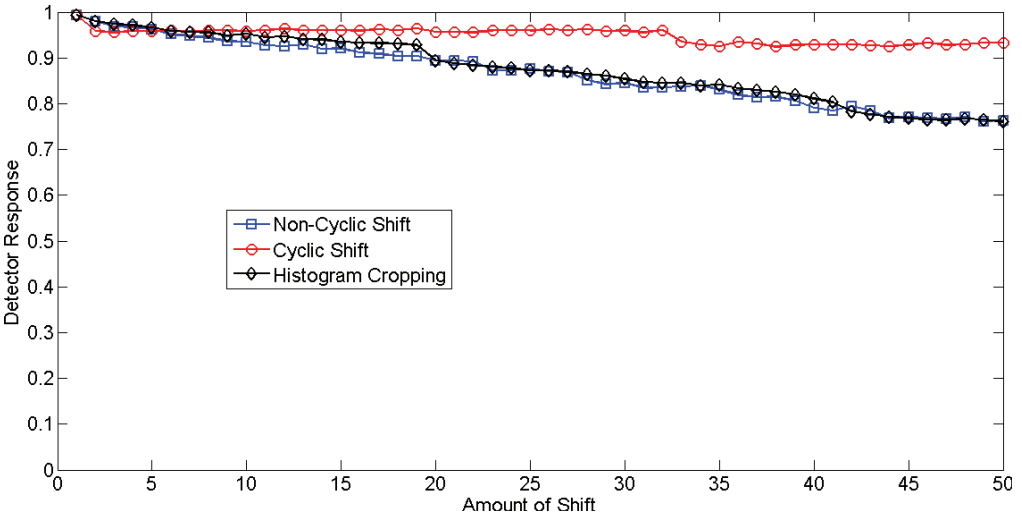


Figure 10. Effects of JPEG and JPEG2000 compression on the blue channel histogram of the Lenna image: (a) Uncompressed image; (b) JPEG compression with quality factor 90 (PSNR 40.78); (c) JPEG2000 compression with compression ratio 10.72 (PSNR 41.70)

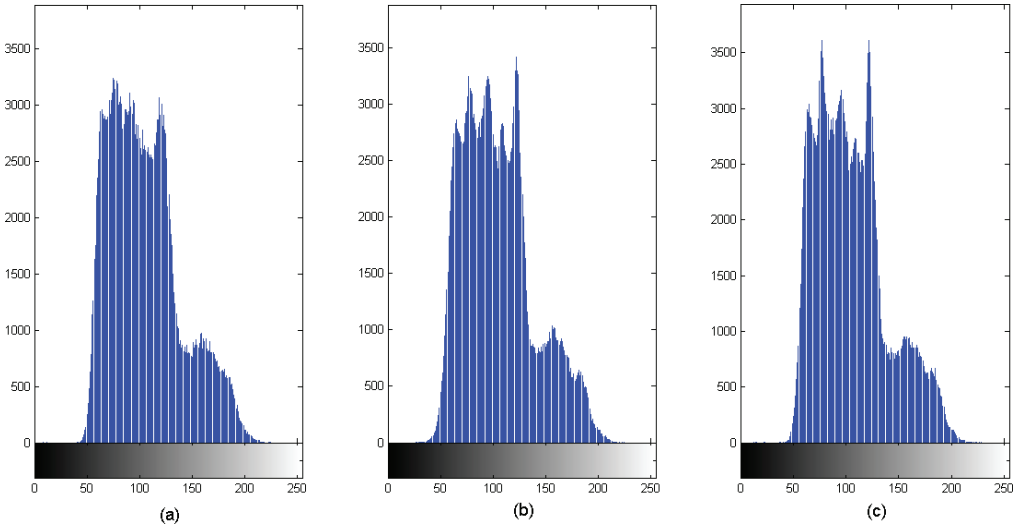
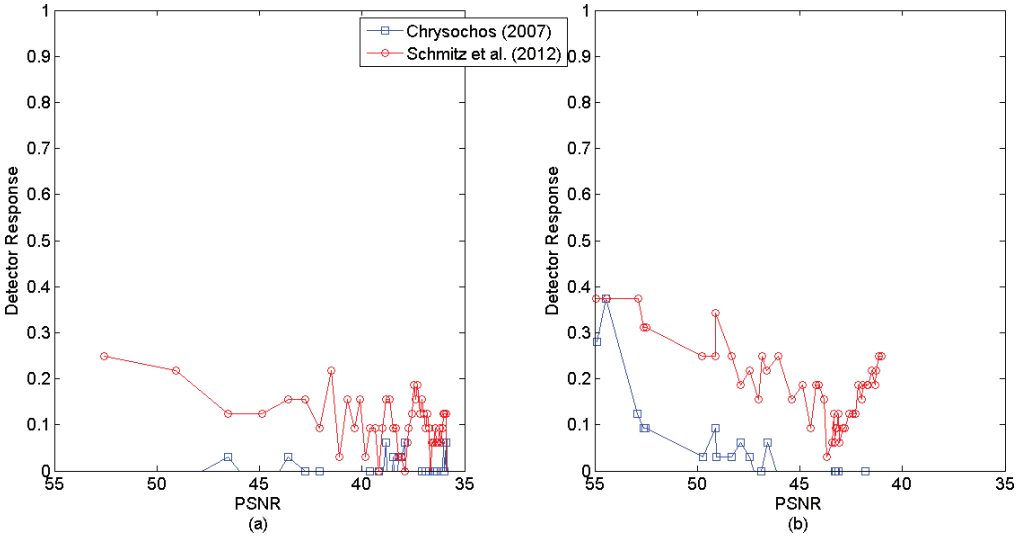


Figure 11. Robustness of earlier histogram-based watermarking algorithms in Lenna image against JPEG and JPEG 2000 compression: (a) JPEG compression; (b) JPEG 2000 compression



embedding is used, i.e. the maximum difference between histogram bin pairs for embedding is enlarged. Other than for the older algorithms, for detection a threshold value of $T = 0.3$ may be used, corresponding to a false positive probability of 0.84% (cf. Sec. 4). This claim is confirmed by Figure 13 showing the averaged robustness results against compression for the 24 images of the Kodak database.

Figure 12. Robustness of proposed CWE algorithm applied to Lenna image: (a) JPEG compression; and (b) JPEG 2000 compression

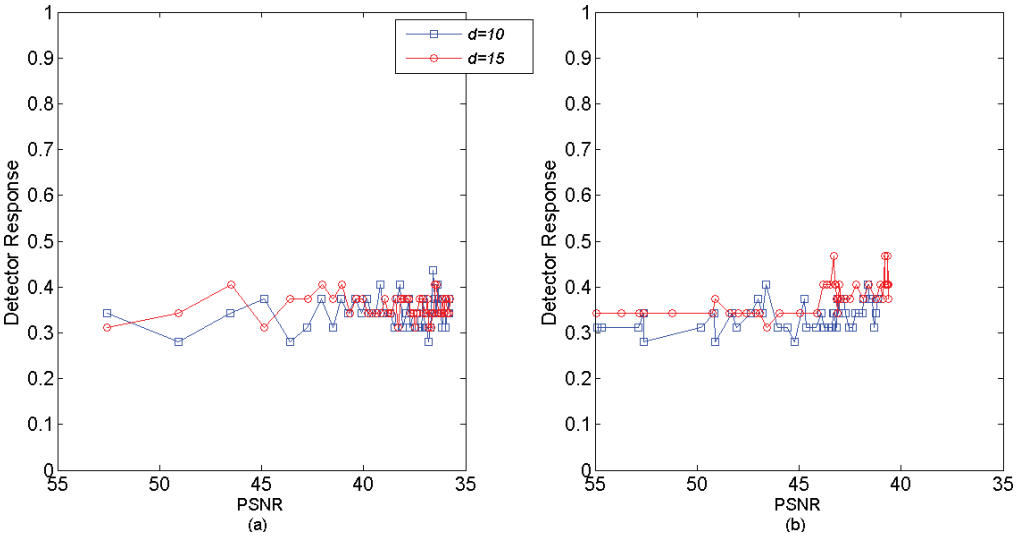
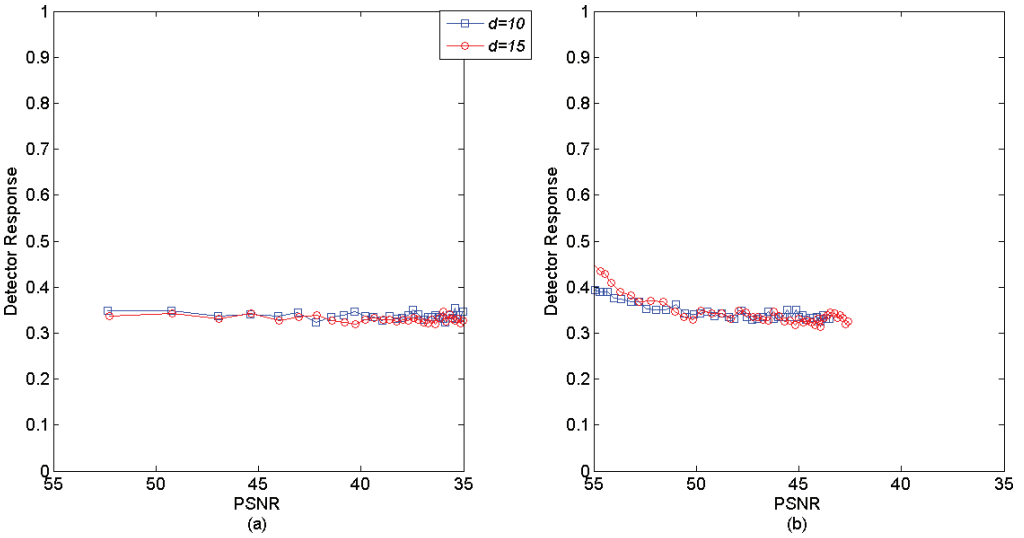


Figure 13. Robustness against JPEG and JPEG2000 compression averaged over 24 test images: (a) JPEG compression; (b) JPEG2000 compression



6. COMPARISON OF RELATED WATERMARKING SCHEMES

Table 1 provides a qualitative overview of the merits of four histogram-based watermarking algorithms, including the one proposed here. The four algorithms were chosen due to the fact that they all base embedding and detection on comparing selected watermarking bins. Robustness against histogram shifts was not

Table 1. Qualitative comparison of histogram based watermarking algorithms

Algorithm	Robustness against Histogram Shifts	Robustness against Lossy Compression	Key Length	Capacity	Suitable for CWE
Chrysochos (2007)	low	low	~ 11 bits	~ 128 bits	yes
Schmitz et al. (2012)	low	low	> 200 bits	~ 128 bits	yes
Xiang et al. (2008)	not tested	high	~ 5 bits	20- 30 bits	no
Proposed	high	moderate	> 200 bits	~ 128 bits	yes

tested for the algorithm in (Xiang et al., 2008), but due to the reported high robustness against lossy compression and the fact that it uses similar techniques to the one proposed here, good robustness against histogram shifts can be expected for that algorithm.

7. CONCLUSION

It is hard to devise a robust watermarking algorithm that can work in the encrypted domain because there are no visually important features to use for embedding in this case. In the present paper, we have extended an earlier algorithm that is commutative with encryption by deploying a synchronization process between the embedder and the detector, making it robust against simple histogram shifts and modestly robust against lossy compression. While Commutative Watermarking-Encryption schemes based on partial encryption have the advantage of higher robustness against lossy compression, the invariant encryption approach proposed here has the advantage of full encryption of the media data. Furthermore, unlike previous more robust histogram-based watermarking algorithms, it is able to use a watermarking key that is long enough to withstand brute-force attacks. Our further work will focus on improving robustness of the presented algorithm against lossy compression and other types of common image processing operations.

REFERENCES

Boho, A., van Wallendael, G., Dooms, A., de Cock, J., Braeckman, G., & Schelkens, P. et al. (2013). End-to-end security for video distribution. *IEEE Signal Processing Magazine*, *30*(2), 97–107. doi:10.1109/MSP.2012.2230220

Chen, B., & Wornell, G. W. (2001). Quantization Index Modulation: A class of provably good methods for digital watermarking and information embedding. *IEEE Transactions on Information Theory*, *47*(4), 1423–1443. doi:10.1109/18.923725

Chen, C., Ni, J., & Huang, J. (2009). Temporal statistic based video watermarking scheme robust against geometric attacks and frame dropping. *Digital Watermarking: 8th. International Workshop IWDW 2009, Guildford, UK, August 24-16, 2009, Proceedings*, pp. 81-95.

Chen, G., Mao, Y., & Chui, C. K. (2004). A symmetric image encryption scheme based on 3D chaotic cat maps. *Chaos, Solitons, and Fractals*, *21*(3), 749–761. doi:10.1016/j.chaos.2003.12.022

Chrysochos, E., Fotopoulos, V., Skodras, A. N., & Xenos, M. (2007). Reversible image watermarking based on histogram modification. *Proc. 11th Panhellenic Conf. Informatics*, pp. 93-104.

Coltuc, D., & Bolon, P. (1999). Robust watermarking by histogram specification. *Proc. Int. Conf. Image Processing*, vol. 2, pp. 236-239.

Cox, I. J., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (2007). *Digital Watermarking and Steganography*. Morgan Kaufman.

Fontaine, C., & Galand, F. (2007). A survey of homomorphic encryption for nonspecialists. EURASIP Journal on Information Security 2007, Article ID 13801.

Herrera-Joancomarti, J., Katzenbeisser, S., Megias, D., Minguillon, J., Pommer, A., Steinebach, M. and Uhl, A. (2005). *First summary report on hybrid systems*, EU project ECRYPT (European Network of Excellence in Cryptology), Deliverable D.WVL.5.

Lagendijk, R.-L., Erkin, Z., & Barni, M. (2013). Encrypted signal processing for privacy protection. *IEEE Signal Processing Magazine*, *30*(1), 82–105. doi:10.1109/MSP.2012.2219653

Lian, S. (2009). Quasi-commutative watermarking and encryption for secure media content distribution. *Multimedia Tools and Applications*, *43*(1), 91–107. doi:10.1007/s11042-008-0258-4

Lian, S., Liu, Z., Zhen, R., & Wang, H. (2006). Commutative watermarking and encryption for media data. *Optical Engineering (Redondo Beach, Calif.)*, *45*(8), 080510. doi:10.1117/1.2333510

Lin, C. H., Chan, D. Y., Su, H., & Hsieh, W. S. (2006). Histogram-oriented watermarking algorithm: Colour image watermarking scheme robust against geometric attacks and signal processing. *IEE Proceedings. Vision Image and Signal Processing*, *153*(4), 483–492. doi:10.1049/ip-vis:20050107

Roy, S., & Chang, E. C. (2004). Watermarking color histograms. *Proc. 2004 Int. Conf. Image Processing*, pp. 2191-2194.

Schmitz, R., Li, S., Grecos, C., & Zhang, X. (2012). A new approach to commutative watermarking-encryption. *Communications and Multimedia Security: 13th IFIP TC 6/TC 11 International Conf., CMS 2012, Canterbury, UK, September 3-5, 2012. Proceedings,* pp. 117-130.

Schmitz, R., Li, S., Grecos, C., & Zhang, X. (2013). Towards more robust commutative watermarking-encryption. *Proceeding of 2013 IEEE International Symposium on Multimedia*, pp. 283-286. doi:10.1109/ISM.2013.54

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, *13*(4), 600–612. doi:10.1109/TIP.2003.819861 PMID:15376593

Xiang, S., Kim, H. J., & Huang, J. (2008). Invariant image watermarking based on statistical features in the low-frequency domain. *IEEE Transactions on Circuits and Systems for Video Technology*, *18*(6), 777–790. doi:10.1109/TCSVT.2008.918843

Xu, Z., Xiong, L., & Xu, Y. (2014). On the provably secure CEW based on orthogonal decomposition. *Signal Processing Image Communication*, *29*(5), 607–617. doi:10.1016/j.image.2013.10.007

CALL FOR ARTICLES

International Journal of Multimedia Data Engineering and Management

An official publication of the Information Resources Management Association

MISSION:

The primary objective **International Journal of Multimedia Data Engineering and Management (IJMDEM)** is to promote and advance multimedia research from different aspects in multimedia data engineering and management. It provides a forum for university researchers, scientists, industry professionals, software engineers and graduate students who need to be become acquainted with new theories, algorithms, and technologies in multimedia engineering, and to all those who wish to gain a detailed technical understanding of what multimedia engineering involves. Novel and fundamental theories, algorithms, technologies, and applications will be published to support this mission.

ISSN 1947-8534
eISSN1947-8542
Published quarterly

COVERAGE/MAJOR TOPICS:

- Content-based retrieval (image, video, audio, etc.)
- Image/video/audio databases
- Learning support for multimedia data
- Multimedia data engineering
- Multimedia data indexing
- Multimedia data mining
- Multimedia data modeling
- Multimedia data storage
- Multimedia databases
- Multimedia systems
- Multimodal data analysis
- Network support for multimedia data
- New standards
- Relevance feedback
- Security support for multimedia data
- Technologies and applications

All inquiries regarding IJMDEM should be directed to the attention of:
Shu-Ching Chen, Editor-in-Chief
ijmdem@igi-global.com
All manuscript submissions to IJMDEM should be sent through the online submission system:
http://www.igi-global.com/authorseditors/titlesubmission/newproject.aspx

Ideas for Special Theme Issues may be submitted to the Editor-in-Chief.

Please recommend this publication to your librarian. For a convenient easy-to-use library recommendation form, please visit:
http://www.igi-global.com/IJMDEM

Coltuc, D., & Bolon, P. (1999). Robust watermarking by histogram specification. *Proc. Int. Conf. Image Processing*, vol. 2, pp. 236-239.

Cox, I. J., Miller, M., Bloom, J., Fridrich, J., & Kalker, T. (2007). *Digital Watermarking and Steganography*. Morgan Kaufman.

Fontaine, C., & Galand, F. (2007). A survey of homomorphic encryption for nonspecialists. EURASIP Journal on Information Security 2007, Article ID 13801.

Herrera-Joancomarti, J., Katzenbeisser, S., Megias, D., Minguillon, J., Pommer, A., Steinebach, M. and Uhl, A. (2005). *First summary report on hybrid systems*, EU project ECRYPT (European Network of Excellence in Cryptology), Deliverable D.WVL.5.

Lagendijk, R.-L., Erkin, Z., & Barni, M. (2013). Encrypted signal processing for privacy protection. *IEEE Signal Processing Magazine*, *30*(1), 82–105. doi:10.1109/MSP.2012.2219653

Lian, S. (2009). Quasi-commutative watermarking and encryption for secure media content distribution. *Multimedia Tools and Applications*, *43*(1), 91–107. doi:10.1007/s11042-008-0258-4

Lian, S., Liu, Z., Zhen, R., & Wang, H. (2006). Commutative watermarking and encryption for media data. *Optical Engineering (Redondo Beach, Calif.)*, *45*(8), 080510. doi:10.1117/1.2333510

Lin, C. H., Chan, D. Y., Su, H., & Hsieh, W. S. (2006). Histogram-oriented watermarking algorithm: Colour image watermarking scheme robust against geometric attacks and signal processing. *IEE Proceedings. Vision Image and Signal Processing*, *153*(4), 483–492. doi:10.1049/ip-vis:20050107

Roy, S., & Chang, E. C. (2004). Watermarking color histograms. *Proc. 2004 Int. Conf. Image Processing*, pp. 2191-2194.

Schmitz, R., Li, S., Grecos, C., & Zhang, X. (2012). A new approach to commutative watermarking-encryption. *Communications and Multimedia Security: 13th IFIP TC 6/TC 11 International Conf., CMS 2012, Canterbury, UK, September 3-5, 2012. Proceedings,* pp. 117-130.

Schmitz, R., Li, S., Grecos, C., & Zhang, X. (2013). Towards more robust commutative watermarking-encryption. *Proceeding of 2013 IEEE International Symposium on Multimedia*, pp. 283-286. doi:10.1109/ISM.2013.54

Wang, Z., Bovik, A. C., Sheikh, H. R., & Simoncelli, E. P. (2004). Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing*, *13*(4), 600–612. doi:10.1109/TIP.2003.819861 PMID:15376593

Xiang, S., Kim, H. J., & Huang, J. (2008). Invariant image watermarking based on statistical features in the low-frequency domain. *IEEE Transactions on Circuits and Systems for Video Technology*, *18*(6), 777–790. doi:10.1109/TCSVT.2008.918843

Xu, Z., Xiong, L., & Xu, Y. (2014). On the provably secure CEW based on orthogonal decomposition. *Signal Processing Image Communication*, *29*(5), 607–617. doi:10.1016/j.image.2013.10.007