Content-Fragile Commutative Watermarking-Encryption Based on Pixel Entropy[⋆]

Roland Schmitz[1], Shujun Li[2], Christos Grecos[3] and Xinpeng Zhang[4]

[1] Stuttgart Media University, Germany
[2] University of Surrey, UK
[3] Independent Image Consultant, Glasgow, UK
[4] Shanghai University, China

Abstract. Content-fragile commutative watermarking-encryption requires that both the content-fragile image signature and the watermarking process are invariant under encryption. The pixel entropy, being dependent on first-order image statistics only, is invariant under permutations. In the present paper we embed semi-fragile signatures based on pixel entropy by using a histogram-based watermarking algorithm, which is also invariant to permutations. We also show how the problem of collisions, i.e. different images having the same signature, can be overcome in this approach, if embedder and encryptor share a common secret.

Keywords: Commutative Watermarking-Encryption, Content-Fragile Watermarking, Pixel Entropy.

1 Introduction

Encryption and watermarking are important techniques for the protection of digital media. While encryption serves to provide confidentiality, watermarks can be used to provide various security services ranging from integrity protection to source authentication and copyright protection. Content-fragile (or semi-fragile) watermarks try to strike the middle ground between exact authentication as provided by cryptographic hash functions or digital signatures, and robust watermarks that are hard to destroy by any image modification. They are supposed to survive benign operations like compression, but should be destroyed by modifications of the image content. In the most common way of content-fragile watermarking, the first step is to compute a content-fragile signature value, which is to represent the semantics of the image. The content-fragile signature is then embedded by some robust watermarking scheme. In the verification process, the

[⋆] This paper has been published in *Advanced Concepts for Intelligent Vision Systems: 16th International Conference, ACIVS 2015, Catania, Italy, October 2629, 2015, Proceedings*, *Lecture Notes in Computer Science*, vol. 9386, pp. 474-485, Springer, 2015.

watermark is extracted from the marked image and compared to the signature value computed from the marked image. Therefore, care must be taken that the watermarking proces does not influence the signature value. Very often, these watermarks are applied separately to small image parts, so that content modifications can be localized.

While there has beem some work in recent years on the problem of combining watermarking and encryption (CWE, see for example [1], [2], [4], [5], [6], and [8]), to the best of our knowledge no content-related watermarks commutative with encryption have been proposed so far. At first glance, however, it seems to be paradoxical to search for a content-fragile watermark that is commutative with encryption. After all, the encryption process is supposed to destroy the visual information from an image, so how can a content-related watermark survive this operation? However, in the past, there have been attempts to define content-fragile signatures which only involve first-order statistics of the image, but no localization information, like the mean histogram value [9] and the pixel entropy [11]. Obviously, this kind of signatures will be invariant under permutation ciphers. The same is true for watermarking strategies that are purely histogram based. In order to be able to combine these two approaches, the watermarking process must not change the histogram in such a way that the content-fragile signature is affected. This paper presents a feasible way to combine a special kind of content-related signature and a watermarking algorithm commutative with permutation ciphers, where the signature is based on the pixel entropy. A common problem with content-related signatures are collisions, i.e., different images having the same content-related signature. Particularly if the signature is based on the histogram alone, collisions are quite easy to find. We show how this problem can be avoided by involving secret information into the signature computation.

The rest of the paper is organized as follows: In Section 2 we discuss some basic properties of the pixel entropy we use in our CWE scheme. Section 3 describes the watermarking process with its three variants: The *localized version* is able to detect and localize content-related changes, the *collision-resistant version* trades localization for collision-resistance, and the *combined version* combines the features of the former two versions. In Section 4 we discuss commutativity of the three versions with encryption, and Section 5 gives some experimental results. Section 6 concludes the paper and gives directions for future research.

2 Properties of Pixel Entropy

In [11], the classical Shannon entropy of an information source [10] is re-defined as pixel entropy of the color channel c of an input image I:

$$PE(I,c) = \sum_{k=0}^{L} p_k \cdot \log_2(p_k), \tag{1}$$

where p_k ia the probability of the grey level k within the color channel c and L is the maximum pixel value. The pixel entropy has three interesting properties which make it useful for content-related commutative watermarking-encryption. Obviously, it is invariant under pixel permutations, meaning it does not change if the image undergoes a permutation-based cipher (**Property 1**). It is also invariant under permutations of the histogram bins of a colour channel (**Property 2**), because permuting the hi)stogram bins will permute the order of summation in (1), but will not change the pixel entropy. This implies that that the pixel entropy is invariant under the watermarking process described in Section 3.

Finally, the sensitivity of the pixel entropy with respect to changes in the grey values of single pixels is governed by the total number N of pixels in the image I (**Property 3**). In order to verify this, we assume that a single pixel has changed its grey value from j to i. Then the new probabilities are $\tilde{p_i} = \frac{n_i+1}{N}$ and $\tilde{p_j} = \frac{n_j-1}{N}$. A direct computation shows that the corresponding change in the pixel entropy is given by

$$\begin{aligned} \mathit{\Delta} PE &= \frac{1}{N}\left(\log\left(\frac{n_i}{n_j}\right) + (n_i+1)\log\left(1+\frac{1}{n_i}\right) + (n_j-1)\log\left(1-\frac{1}{n_j}\right)\right) \\ &\approx \frac{1}{N}\left(\log\left(\frac{n_i}{n_j}\right) + \frac{1}{n_i} + \frac{1}{n_j}\right), \qquad (2) \end{aligned}$$

as $\log(1+x) \approx x$ for small x.

As mentioned above, content-related signatures can be applied to small subimages to localize content modifications. If the pixel entropy is used for this purpose, the size of the subimages should be minimized for maximum sensitivity, because we have no control over n_i or n_j.

Note that invariance under permutation ciphers does not hold for the second order pixel entropy,

$$PE^2(I,c) = \sum_{i=0}^{L}\sum_{j=0}^{L} p_{ij} \cdot \log_2(p_{ij}), \qquad (3)$$

where p_{ij} is the probability of ocurrence of the pair (i,j) of grey values within the colour channel c, because PE^2 is also a function of the pixel's scanning order.

3 Watermarking Process

The basic watermarking process closely follows the approach taken in [8] and [7] and is based on the idea of swapping histogram bins according to a secret watermarking key W_K [3]: For each watermark bit $w_i \in \{-1,1\}$, the algorithm randomly selects a certain histogram bin a_i and another bin b_i within a d-neighbourhood of a_i, taking W_K as initial seed. Here, d is a fixed parameter governing the tradeoff between robustness and transparency of the watermark. Histogram bins of equal height are not selected. Now, 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. If this is not the case, the two bins are swapped. For watermark extraction, the bins at the positions specified by W_K are compared. If a reference watermark is known before extraction, the authenticity of the image can be verified by computing the linear correlation of the reference mark and the extracted mark.

Here, a reference mark m is provided by the pixel entropies of the color channels. For embedding purposes, each pixel entropy is turned into an integer by first multiplying it by 10^4 and then quantizing it with a quantization factor q. More specifically, for a given input image I and colour channel c, the signature is calculated as follows:

$$m(I,c) = \left\lfloor \frac{10^4 \times PE(I,c)}{q} \right\rceil \times q \qquad (4)$$

In the following embedding examples q has been set to 2. The 16 most significant bits of $m(I,c)$ are converted into a 16-bit bipolar bitstring $w(I,c)$ and embedded into the corresponding colour channel c by the procedure described above. It is important to note that this procedure does not affect the pixel entropy according to Property 2 given in Section 2. After detection, the three extracted 16-bit integers are concatenated and compared to the concatenated pixel entropies found in the color channels of the marked image by computing the linear correlation. An image is deemed unauthentic, if the linear correlation is below a certain threshold T. Assuming a balanced distribution of the bipolar bits in $w(I,c)$, the corresponding False Positive probability is [7]:

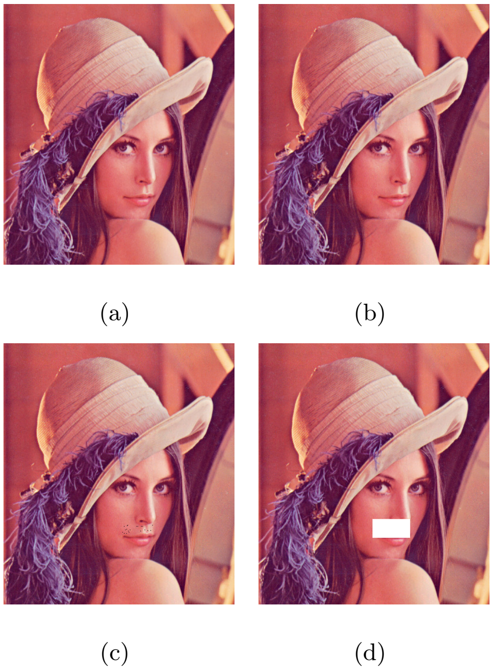
$$p(\text{False Positive}) = \left(\frac{1}{2}\right)^N \cdot \sum_{k=\lceil \frac{N}{2}(T+1)\rceil}^{N} \binom{N}{k}, \qquad (5)$$

where N is the length of the embedded mark. In what follows, we have set $T = 0.8$, while $N = 3 \times 16 = 48$ as discussed above. This gives a false positive probability of 7.57×10^{-10}. For a grey-scale image, we have $N = 16$ and a false positive probability of 2.59×10^{-4}. False negative probabilities, on the other hand, are notoriously difficult to estimate as it is not clear which attacks are performed on a marked image.

In what follows, we describe three variants of the general watermarking scheme described above. While in all variants certain subimages are marked, the variants differ in the way the subimages are formed.

3.1 Localized Version

The aim of the localized version is to identify subimages where image modifications have taken place. To this end, square subimages with a side length s are formed in a regular, non-overlapping fashion, then marked with their respective pixel entropies. Experience has shown that the minimum subimage size that can provide meaningful histograms and a sufficient embedding capacity is $s = 32$. This size therefore offers the most fine-grained localization of changes and the highest sensitivity to changes. As an example, the upper row of Figure 1 shows a plaintext image and the corresponding marked image in the localized version with a subimage size $s = 32$. The visual quality of the marked image is assessed by computing the Peak Signal to Noise Ratio (PSNR) and the Structural Similarity Index (SSIM, [12]). Next, the marked Lenna image was modified setting the grey values of 20 random pixels in the area between upper lip and nose to 0. Figure 1 (c) and (d) show the modified image and verified image, indicating the two subimages where the modifications have taken place. Specifically, in Figure 1(d), a regular subimage is rendered white if one of the random sub-subimages yields a correlation value $T < 0.8$.



(a) (b)

(c) (d)

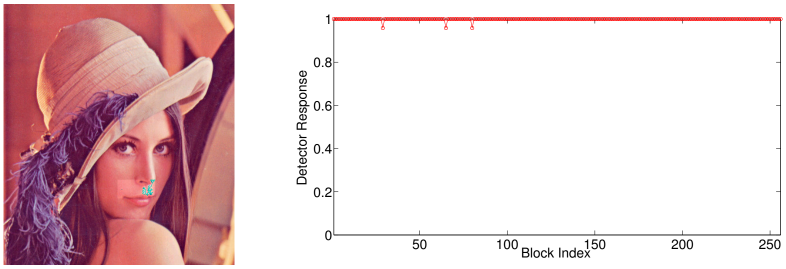
Fig. 1. (a) Original image; (b) Marked image (PSNR 43.82 db, SSIM 0.98); (c) Modified marked Lenna image; (d) Verified Lenna image.

3.2 Collision-Resistant Version

It is relatively easy to generate collisions, i.e. to generate different images that have the same pixel entropy. In principle, any permutation of the histogram bins will produce an image version with the same pixel entropy as the original image. While most of these permutations will destroy the watermark because embedder and detector are de-synchronized, the watermarking process can be made robust against cyclic shifts by a suitable calibration of the embedding and detection process [7]. In a cyclic shift of the histogram, the grey values of the pixels in the three colour channels undergo the following transformation:

$$P_{\text{attacked}}(i,j) = (P(i,j) + x) \mod 256, \tag{6}$$

where x is a positive or negative integer. Due to the wrap-up at the end of the histogram, cyclic histogram shifting may lead to visible changes of the image content, as Figure 2 shows, where two subimages of the Lenna image have been cyclically shifted by an amount of $x = 20$. The localized version introduced in the last subsection is unable to detect the different image contents, because of the unchanged pixel entropy of the subblocks and the robustness of the underlying watermarking process.

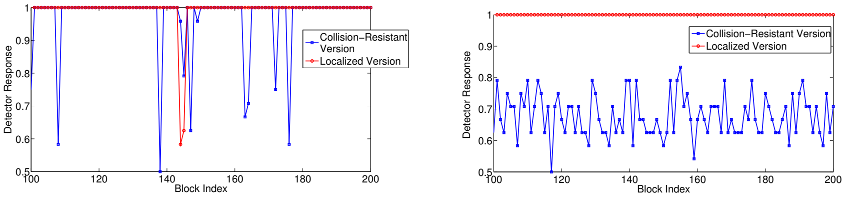


(a) (b)

Fig. 2. (a) Two subimages of the marked Lenna image have been cyclically shifted; (b) Cyclic shifting goes unnoticed by the detector in the localized version.

In order to cope with the collision problem, some secret information needs to be introduced into the pixel entropy computation. More specifically, let I be an original image of size $H \times W$. We generate an $m \times n$ array of subimages of size s, where $m = H/s$, $n = W/s$, by pseudo-randomly choosing pixels from the original image and assigning them to the subimages in turn, under control of a secret `SplitKey`. The resulting subimages are marked separately with their pixel entropy. After marking, the subimages are merged back to form the watermarked image. The complete Split-and-Merge process is described in Section 4 in greater detail, where we investigate its interplay with encryption.

Through the Split process, any local change in the watermarked image will be randomly distributed over the subimages and lead to corresponding changes of the pixel entropy, which cannot be foreseen by an attacker, unless she knows `SplitKey`. This fact is illustrated in Figure 3, where we compare the correlation values of the subimages of Figures 1(c) and 2(a) for the localized version and the collision-resistant version. Because of the distribution over subimages, however, localization information of changes is lost in this approach. Here, we have used $m = n = 2$, but different values for m and n are possible as well, especially if the original image is not square. In any case, s must be common divisor of H and W (see Section 5).



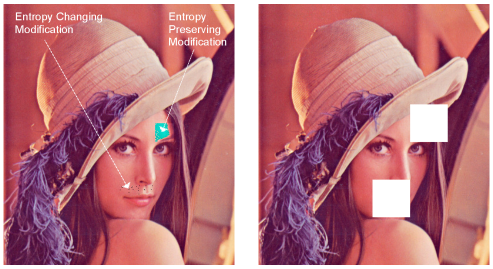
(a)

(b)

Fig. 3. (a) Detector Responses for modified Lenna image 1(c); (b) Detector Responses for modified Lenna image 2(a).

3.3 Combined Version

It is possible to combine the virtues of the localized and the collision-resistant version by splitting the image in a regular fashion first and applying the collision-resistant approach to the resulting subimages. This means that the subimages are split again randomly into a 2×2 array of sub-subimages. If we maintain the minimum size of $s = 32$ for the irregular sub-subimages, the regular subimages have a minimum size of $s_{reg} = 64$ in the combined version. The modified Lenna image shown in Figure 4(a), for example, has been split into an 8×8 array of regular subimages of size $s_{reg} = 64$ each. The subimages were further split into an 2×2 array of randomly formed sub-subimages which were watermarked afterwards. Both entropy-preserving and non-entropy preserving image modifications can be localized on subimage level (see Figure 4).


Entropy Changing Modification
Entropy Preserving Modification

(a)

(b)

Fig. 4. (a) Modified marked Lenna image; (b) Verified Lenna image (Combined Version).

4 Commutativity with Encryption

Commutativity of the marking process with encryption means that

$$\mathcal{M}(\mathcal{E}_K(I), m) = \mathcal{E}_K(\mathcal{M}(I, m)) \tag{7}$$

holds, where $\mathcal{E}$ is the encryption function, K is the encryption key, I is the plaintext media data, $\mathcal{M}$ is the marking function and m is the mark to be embedded. In the next subsections, we discuss the issue of commutativity with encryption for the three watermarking variants introduced in Section 3.

4.1 Localized Version

The localized version is commutative to encryption, if the encryption function $\mathcal{E}$ is permutation based and is confined to the same regular subimages of size s as the watermarking process (see Fig. 5 (a)). In this case, encryption and watermarking commute at subimage level, because the watermarking process is purely histogram-based and does not include any components which are changeable by applying a permutation.

4.2 Collision-Resistant and Combined Version

The collision-resistant version splits the whole image randomly into subimages, while in the combined version first regular subimages are formed, which are split randomly afterwards. As both methods differ only in the size of subimages they are applied to (actually, the collision-resistant version can be seen as an instance of the combined method using a single large subimage), we can restrict our discussion to the collision-resistant method.

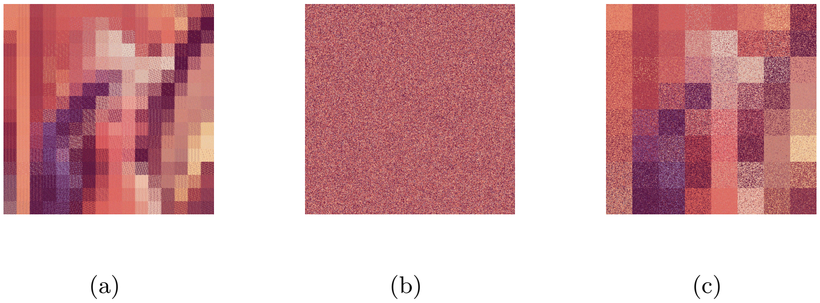
As in the localized version, one has to make sure, that the watermarking and encryption processes act on the same subimages. To this end, both marking function $\mathcal{M}$ and encryption function $\mathcal{E}$ have to use a common secret `SplitKey` which governs the random split process. More specifically, the encryption process has to include the same Split-and-Merge cycle as the marking process. As embedding and watermarking are not completely independent of each other anymore in this case, they may be called *quasi-commutative* instead.

To check whether encryption and watermarking actually commute if they share `SplitKey`, let us go through the encryption and marking processes for an original image I in detail:

- **Encrypt-then-Mark:**
  A pixel P at position (r_I, c_I) within I with a pixel value g is selected by the random number generator and assigned to row r_A, column c_A in subimage A. Permuting subimage A sends the pixel to a different position (r_e, c_e) within A. The merging process will asssign this pixel to some position (i, j) within I, thus $\mathcal{E}(I)(i, j) = g$. When marking $\mathcal{E}(I)$, the split process will first assign position (i, j) to position (r_e, c_e) within subimage A. Marking A may change the pixel value to g_M, and merging back yields $\mathcal{M}(\mathcal{E}(I))(i, j) = g_M$.

- **Mark-then-Encrypt:**
  The Split process assigns P to position (r_A, c_A) within subimage A. Marking changes P'spixel value to g_M. Merging back into I gives $\mathcal{M}(I)(r_I, c_I) = g_M$. Encryption assigns this pixel first to (r_A, c_A) within A by the Split process and then to (r_e, c_e) by permuting A. Merging back into I yields $\mathcal{E}(\mathcal{M}(I))(i, j) = g_M$.

Figure 5 shows the results of appliying a permutation cipher to the marked subimages for the three described versions of the marking process, i.e. the right-hand side of Eq. 7. The mark can be extracted from the encrypted image in the same way as from the plaintext image. In the collision-resistant and combined versions, however, `SplitKey` has to be known to the detector as well.



(a) (b) (c)

Fig. 5. Marked, then encrypted Lenna image: (a) Localized version; (b) Collision resistant version; (c) Combined version.

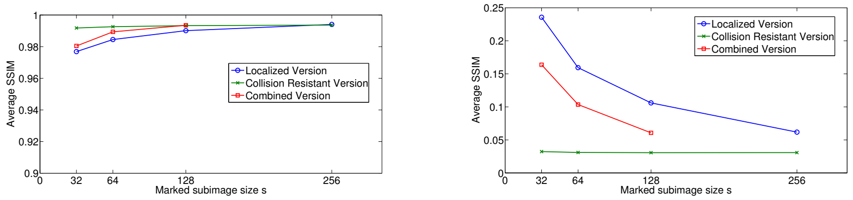
Note that although only the marked subimages are permuted, the random process used for splitting into subimages in Figures 5 (b) and (c) distributes the permutation further into the image. It is clearly visible that there is a tradeoff between cipher security on one hand (both in terms of key space and opacity of image features) and sensitivity and the ability to localize image changes on the other.

5 Experimental Results

In this section we report the results of applying the three versions of the watermarking process to a set of 25 test images with three different formats, namely 512×768, 768×512 and 512×512. One of the test images is the Lenna image, the other 24 come from the Kodak true-color image database. As the subimage size must be a common divisor of height and width of the images, we investigated the influence of the possible subimages sizes $s \in \{32, 64, 128, 256\}$ on transparency of the watermark, opacity of the marked, encrypted images and on sensitivity of the mark with respect to pixel value changes within marked images.

5.1 Visual Quality of Watermarked Images and Effectivity of Encryption

For Fig. 6, the 25 images of the test set were watermarked by the three variants, using different values for the subimage size s. The plot shows the resulting SSIM values, computed by comparing the original image to the watermarked image and the watermarked encrypted image, respectively. The plotted values have been averaged over the 25 images in the test set.



(a) (b)

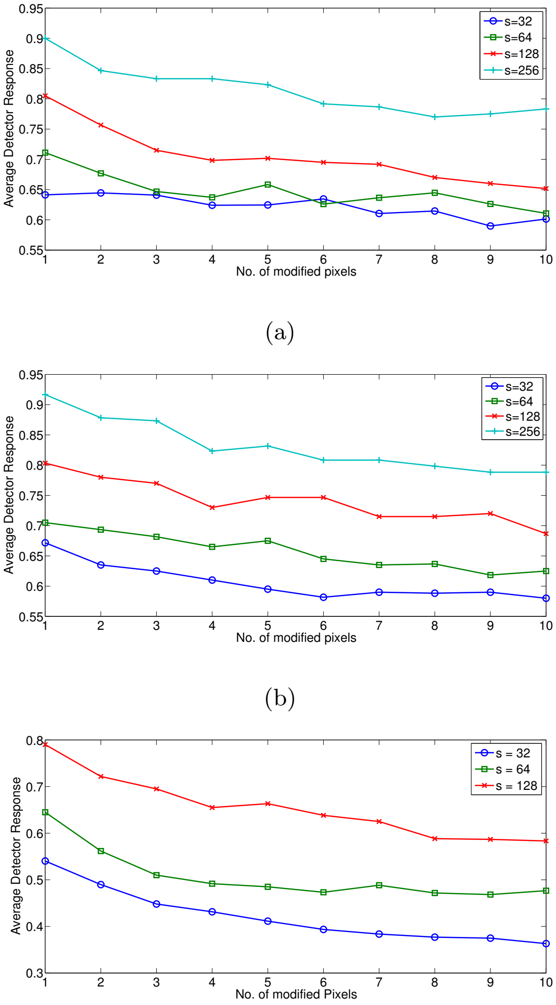
Fig. 6. Average SSIM values for (a) watermarked images and (b) watermarked encrypted images with different subimage sizes

All three methods have a very similar visual quality, while the additional randomization of changes in the collision resistant seems to affect the perceptibility of the watermark in a slighty positive way (see Figure 6(a)). Figure 6(b) confirms the expectation that greater subimage sizes lead to a better concealment of visual features, as the permutations are applied to larger subimages. In the collision resistant method, the random split process acts like an additional permutation of the complete image. Therefore, the subimage size does not affect the effectiveness of the cipher in this case.

5.2 Sensitivity to Local Changes

Figure 7 shows the average detector responses for the localized and the combined watermarking methods, using different s-values. More specifically, the minimal detector response over all subimages is recorded for each image and then averaged over all images. Here, a watermarked image was modified by altering a number of pixels p, where $1 \leq p \leq 10$. In order to make the results as comparable as possible, the modifications were always done within the same 32×32 pixel subimage.

As predicted theoretically, the sensitivity is generally higher for smaller subimage sizes, i.e., the average correlations are lower for smaller subimage sizes. Overall, the three methods behave very similarly. For very large subimage sizes, the watermarking variants are not sensitive enough to small changes to be able to reliably detect those changes.



(a)

(b)

(c)

Fig. 7. Average detector responses for watermarked and modified images: (a) Localized Method; (b) Collision Resistant Method; (c) Combined Method

6 Conclusion

We have presented a way to realize content-related watermarking that is commutative with permutation based encryption. Both the content-related signature and the watermark are based on first-order statistics and are therefore invariant under permutation ciphers. Moreover, the watermarking process does not influence the signature. The watermark can therefore embedded into the same subimages which are authenticated by it. We have identified some fundamental tradeoffs between collision resistance and cipher security hand and the ability to locate content changes within the image. Our further research will focus on the question of robustness of the presented scheme against benign, content-preserving operations in the plaintext domain.

References

1. Battisti, F., Cancellaro, M., Boato, G., Carli, M., Neri, A.: Joint watermarking and encryption of color images in the Fibonacci-Haar domain. EURASIP J. Advances in Signal Processing 2009, Article ID 938515 (2009)
2. Boato, G., Conotter, V., De Natale, F.G.B., Fontanari, C.: A joint asymmetric watermarking and image encryption scheme. In: Security, Forensics, Steganography, and Watermarking of Multimedia Contents X. Proc. SPIE, vol. 6819, p. 68191A (2008)
3. Chrysochos, E., Fotopoulos, V., Skodras, A.N., Xenos, M.: Reversible image watermarking based on histogram modification. In: Proc. 11th Panhellenic Conf. Informatics (PCI 2007). pp. 93–104 (2007)
4. Guo, J., Zheng, P., Huang, J.: Secure watermarking scheme against watermark attacks in the encrypted domain. Journal of Visual Communication and Image Representation 30(0), 125 – 135 (2015)
5. Lemma, A., Katzenbeisser, S., Celik, M., van der Veen, M.: Secure watermark embedding through partial encryption. In: Proc. 5th Int. Workshop Digital Watermarking (IWDW 2006). Lecture Notes in Computer Science, vol. 4283, pp. 433–445 (2006)
6. Lian, S.: Quasi-commutative watermarking and encryption for secure media content distribution. Multimedia Tools and Applications 43(1), 91–107 (2009)
7. Schmitz, R., Li, S., Grecos, C., Zhang, X.: Towards more robust commutative watermarking-encryption of images. In: Multimedia (ISM), 2013 IEEE International Symposium on. pp. 283–286 (Dec 2013)
8. Schmitz, R., Li, S., Grecos, C., Zhang, X.: A new approach to commutative watermarking-encryption. In: De Decker, B., Chadwick, D. (eds.) Communications and Multimedia Security. Lecture Notes in Computer Science, vol. 7394, pp. 117–130. Springer Berlin Heidelberg (2012)
9. Schneider, M., Chang, S.F.: A robust content based digital signature for image authentication. In: Image Processing, 1996. Proceedings., International Conference on. vol. 3, pp. 227–230 (Sep 1996)
10. Shannon, C.E.: A mathematical theory of communication. Bell System Technical Journal 27, 379–423 (1948)
11. Thiemert, S., Sahbi, H., Steinebach, M.: Using entropy for image and video authentication watermarks. In: Proc. SPIE. vol. 6072, pp. 607218–1 – 607218–10 (2006)

12. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. Image Processing, IEEE Transactions on 13(4), 600–612 (2004)