applied sciences

MDPI

Article

Color Image Self-Recovery and Tampering Detection Scheme Based on Fragile Watermarking with High Recovery Capability

Rogelio Reyes-Reyes, Clara Cruz-Ramos, Volodymyr Ponomaryov *, Beatriz P. Garcia-Salgado and Javier Molina-Garcia

Instituto Politecnico Nacional, ESIME Culhuacan, Santa Ana 1000, Mexico-City 04440, Mexico; rreyesre@ipn.mx (R.R.-R.); ccruzra@ipn.mx (C.C.-R.); bgarcias1404@alumno.ipn.mx (B.P.G.-S.); javier.molina.21016@gmail.com (J.M.-G.)

* Correspondence: vponomar@ipn.mx

Abstract: In this paper, a fragile watermarking scheme for color image authentication and self-recovery with high tampering rates is proposed. The original image is sub-sampled and divided into non-overlapping blocks, where a watermark used for recovery purposes is generated for each one of them. Additionally, for each recovery watermark, the bitwise exclusive OR (XOR) operation is applied to obtain a single bit for the block authentication procedure. The embedding and extraction process can be implemented in three variants (1-LSB, 2-LSB or 3-LSB) to solve the tampering coincidence problem (TCP). Three, six or nine copies of the generated watermarks can be embedded according to the variant process. Additionally, the embedding stage is implemented in a bit adjustment phase, increasing the watermarked image quality. A particular procedure is applied during a post-processing step to detect the regions affected by the TCP in each recovery watermark, where a single faithful image used for recovery is generated. In addition, we involve an inpainting algorithm to fill the blocks that have been tampered with, significantly increasing the recovery image quality. Simulation results show that the proposed framework demonstrates higher quality for the watermarked images and an efficient ability to reconstruct tampered image regions with extremely high rates (up to 90%). The novel self-recovery scheme has confirmed superior performance in reconstructing altered image regions in terms of objective criteria values and subjective visual perception via the human visual system against other state-of-the-art approaches.

Keywords: fragile watermarking; image self-recovery; image authentication; image tampering detection; tampering coincidence problem

Citation: Reyes-Reyes, R.; Cruz-Ramos, C.; Ponomaryov, V.; Garcia-Salgado, B.P.; Molina-Garcia, J. Color Image Self-Recovery and Tampering Detection Scheme Based on Fragile Watermarking with High Recovery Capability. *Appl. Sci.* **2021**, *11*, 3187. https://doi.org/10.3390/app11073187

Academic Editor: Mauro Castelli

Received: 4 March 2021
Accepted: 30 March 2021
Published: 2 April 2021

1. Introduction

Currently, the development of authentication and reconstruction techniques for digital images has been the focus of extensive research due to the accelerated growth of image editing software, which can be used to tamper with digital images in multiple ways. These authentication and reconstruction techniques are used to detect tampered regions in images where, in the case of alteration, a recovery process should be applied to retrieve the original content. These schemes are helpful in different applications, in which undetected modifications of digital images may have serious consequences, e.g., legal proceedings, where a digital image can be used as legal evidence. Therefore, detection and recovery of tampered content in digital images have become issues of outstanding importance.

In recent years, watermarking techniques have been used to authenticate and recover tampered information in digital images [1–24]. Watermarking techniques can be classified into three types [1,2]: fragile, semi-fragile and robust. Fragile watermarking [1–19,23,24] does not support intentional and unintentional attacks; so, in case of any modification, the watermark would be destroyed. In contrast, these techniques offer a high payload capacity and are mainly used for authentication [3–7] that justifies several frameworks, which

recently appeared in the self-recovery of tampered image regions [1,2,8–19,23,24]. Semi-fragile watermarking techniques are commonly used for copyright protection [20–22] and recovery schemes [25–30]. These techniques are designed to resist non-intentional manipulations caused by traditional image processing operations such as JPEG compression and scaling. These methods are fragile against intentional manipulations like tampering, resulting in a lower recovery rate compared to strategies based on fragile watermarking. Finally, robust watermarking techniques [31–33] are mainly used for copyright protection because they support intentional and non-intentional attacks. Their main disadvantage is a reduced payload capacity in comparison with fragile and semi-fragile watermarking techniques.

Self-recovery techniques based on watermarking consist of initially using small blocks of an image; subsequently, a watermark should be generated and embedded into a different block for content recovery for each self-recovery block. During the recovery process, all tampered blocks are reconstructed by the recovered watermark. This step could fail when a block containing the recovery watermark has been tampered with. In this way, it is impossible to reconstruct a concrete block, generating the so-called *tampering coincidence problem*.

Considering the approaches mentioned above for authentication and tamper detection, which are based on watermarking, the following properties are required for efficient implementation:

(a) *A minimum number of bits* used for recovery and tamper detection: the recovery bits should be embedded redundantly, thus avoiding the tampering coincidence problem.
(b) *Watermark imperceptibility:* the embedded recovery and authentication bits must not affect the visual quality of a watermarked image.
(c) *Precise tamper detection*: a majority of intentional modifications should be accurately detected.
(d) *Precise recovery of tampered regions:* the reconstructed image must demonstrate acceptable visual and objective quality in the reconstructed areas.

In this paper, a *self-recovery of high tampering rate* framework (denoted as *SR-HTR)* is designed according to the previously presented properties required for an efficient authentication scheme and tamper detection in color images. This novel fragile scheme appears to demonstrate a high payload capacity that can be used for authentication and recovery processes. To minimize the negative influence of the tampering coincidence problem, the proposed algorithm generates $15AB/16$ bits in total, $A \times B$ being the size of the image. Additionally, $AB/16$ bits should be produced to detect tampered pixels in any area of an image. Thus, the designed framework can embed three, six or nine copies of recovery and authentication watermarks, achieving a high recovery and tamper detection capability. The novel framework can be implemented using several variants for embedding the recovery and authentication watermarks, such as the *least-significant bit* (LSB) methods: 1-LSB, 2-LSB and 3-LSB, where each one of them provides different advantages that are analyzed below.

For the 2-LSB and 3-LSB embedding processes, a bit adjustment stage is performed [34] on the watermarked pixels, increasing the protected images' objective quality. A hierarchical algorithm in tamper detection is employed to achieve higher tamper detection accuracy. Additionally, an inpainting process is used to resolve the tampering coincidence problem by regenerating the eliminated blocks.

For evaluating the quality of the results obtained in the numerous experiments, the *peak signal-to-noise ratio* (PSNR) and *structural similarity index measure* (SSIM) criteria are employed in this study. Moreover, we use a variation of the PSNR criterion, denoted as PSNR-HVS-M, which considers the *human visual system* (HVS) and visual masking (M). This criterion employs the *contrast sensitivity function* (CSF) and maintains a close relationship with *discrete cosine transform* (DCT) basis functions [35]. Additionally, it has demonstrated good correspondence with human subjective visual perception. Consequently, it could be useful for the justification of the good performance of our novel system.

The rest of this paper is organized as follows. Firstly, Section 2 presents a review of related works. Secondly, Section 3 describes the proposed SR-HTR, followed by Section 4, which explains the experimental setup. Section 5 shows the proposed method's analy-

sis when the embedding and extraction processes in 1-LSB, 2-LSB and 3-LSB are used. Section 6 presents the experimental results obtained by the proposed framework and their performance comparison against state-of-the-art techniques. Finally, the study's conclusion is drawn in Section 7.

2. Related Works

In this section, previously reported watermarking algorithms for authentication and self-recovery in digital images are discussed.

Singh et al. [2] divided the image to be protected into non-overlapping blocks of 2×2 pixels where, for each block, two flags are generated for tamper detection. In addition, the DCT and a quantization procedure are applied to each DCT block. Next, the most important coefficients are identified and, finally, they are transformed to 10 bits. The LSB method is used to embed the recovery and authentication bits.

Zhang et al. [8] proposed an algorithm for authentication and self-recovery based on a fragile watermarking scheme, in which the recovery bits should be obtained from the original DCT coefficients and a compressive sensing approach. The watermark does not contain additional redundancy, and the embedding process is based on the LSB method, in which the watermark data replace the three LSBs of each pixel.

He et al. [9] generated the recovery bits using non-overlapping blocks of 2×2 pixels. Six recovery bits are generated for each block using six *most significant bits* (MSBs) by averaging and normalization processes. The final watermark is obtained by an encryption process of the six recovery bits and a secret key, where the recovery bits are used to locate the tampered regions. Then, two LSBs of each pixel are used for watermark embedding.

Tong et al. [10] proposed a scheme where two flags are obtained for each non-overlapping block of 2×2 pixels to generate tamper detection bits. In addition, the recovery bits are generated according to the average intensity of each block. The embedding technique uses the LSB method, where the three LSBs are utilized for watermark embedding.

Qian et al. [11] performed non-overlapping block divisions on an image. These blocks are classified into four types according to their characteristics. For each classified block, a quantization process is applied to the DCT coefficients. These quantized coefficients are embedded as a recovery watermark, and a hash function MD5 obtains the authentication bits. The embedding process is performed by the LSB method. Finally, to recover the corrupted regions after the recovery process, an inpainting technique is applied.

Qin et al. [12] divided the image into non-overlapping 8×8 pixel blocks, classified into textured and smooth blocks. The recovery bits are obtained using *vector quantization* (VQ) indexing and inpainting processes, where VQ indexing is used for each complex block. Finally, both VQ indexing and inpainting are used for each smooth block.

Li et al. [13] proposed a scheme for the protection of biometric images based on salient region detection, where each biometric image is divided into *salient regions* (ROIs) and *background regions* (ROBs). Additionally, each image is divided into blocks of 4×4 pixels, and for each block, 16-bit authentication is performed using MD5. Eigenface coefficients obtain the self-recovery watermark, and the embedding process is based on the spatial domain, where operations between the LSB bits and watermark bits are performed.

Jie et al. [14] divided an image into non-overlapping blocks of 2×2 pixels, where six recovery bits and two key-based data bits are produced using an averaging process of the MSB pixels. Additionally, to improve the recovered image quality, a 3×3 block neighborhood is used to recover each tampered block whose feature hidden in another block is corrupted. The number of blocks in a 3×3 block neighborhood of the test block that is inconsistent with their mapping block is obtained to validate an image block. The embedding process is performed by the LSB method.

Dadkhah et al. [15] used *singular value decomposition* (SVD) to generate bits for tamper detection and self-recovery. The image is divided into non-overlapped blocks of 4×4 pixels, and these blocks are divided into blocks of 2×2 pixels. For each block of 4×4 pixels, 12 bits of authentication are generated, and for each block of 2×2 pixels, 20 bits are generated for the recovery process. The embedding scheme is based on the LSB method, where watermarks are embedded into the two LSB bits of each pixel.

Fan [23] proposed an improvement of the reconstruction scheme based on watermarking and the *set partitioning in hierarchical trees* (SPIHT) transform, where this scheme uses blocks of 32×32 pixels to generate the reconstruction bits. Additionally, this algorithm incorporates two versions of the recovery bits to obtain better robustness against the tampering coincidence problem. This scheme's main disadvantage is that the SPIHT transform can result in wrong recovery bits extracted for a block. Therefore, a complete block can be recovered with inferior quality, resulting in poor objective quality of the reconstructed image.

Tai et al. [24] proposed a scheme based on the *integer wavelet transform* (IWT), where the recovery and authentication bits are generated using this transformation, and the embedding scheme is based on LSB. This scheme does not embed the redundancy in data, and as a result, suffers the drawback that it cannot avoid the tampering coincidence problem.

Chamlawi et al. [25] proposed a system based on a semi-fragile watermark in the *discrete wavelet transform* (DWT) domain. The generation of watermarks is performed by DWT–DCT, where the *low frequency* (LL) sub-band is obtained, followed by the application of DCT for non-overlapped blocks. Finally, a quantization step to get the self-recovery watermark is used. The authentication watermark is a binary watermark correlated with the LL sub-band of the first DWT decomposition level. This method's principal drawback is its low self-recovery rate because the embedding stage for the recovery bits is performed at the second decomposition level of the DWT in the middle-frequency sub-bands.

In [26], a semi-fragile watermark was employed using the IWT domain, where self-recovery bit generation was performed utilizing *integer discrete cosine transformation* (IDCT), Huffman coding and an error-correcting procedure based on the *Bose–Chaudhuri–Hocquenghem* (BCH) encoder. The authentication bits were obtained using the exclusive addition operation between the LL frequency sub-band and a pseudo-random sequence.

Chamlawi et al. [27] used a semi-fragile watermark employing the IWT, where the authentication bits are embedded in the LL sub-band; in the opposite, the self-recovery bits are embedded in the *high-low* (HL) and *low-high* (LH) frequency sub-bands. The self-recovery bits are generated by IWT, DCT and a quantization process to obtain the DCT coefficients. Additionally, the tamper detection bits are caused by a bitwise exclusive OR operation between a pseudo-random sequence and a binary matrix.

In a recent study [36], we have proposed a novel method performed on an image in *luma and chroma* (YCbCr) color space, implementing the halftoning algorithm in the luminance channels, thus obtaining recovery bits. Additionally, this scheme uses the procedure proposed in [15], where for chrominance channels, the watermark bits are also generated for recovery purposes. Additionally, three copies of these bits were embedded using the 2-LSB method, incrementing the robustness, and in such a way resolving the tampering coincidence problem. Contrarily, this study's proposition consists of the SR-HTR method, which uses a block-based method for the generation of recovery and authentication watermarks employing an RGB color image. The novel framework is designed in three variants (1-LSB, 2-LSB and 3-LSB), where for each variant, there can be embedded three, six and nine copies of the recovery watermark, respectively. In opposition to the scheme proposed in this study, the framework designed in [36] generates a bit rate of 1.75 *bits per pixel* (*bpp*) for recovery, and 0.25 *bpp* for authentication purposes that, as can be seen below, permit one to obtain a sufficiently good performance in the reconstruction of image areas with tampering rates from 10% to 40%. The principal limitation of the framework [36] consists of embedding up to three copies of the recovery watermark because it generates 4 bits of authentication for each block of 4×4 pixels. Therefore, it cannot obtain good objective and subjective performance in color image reconstruction for high tampering rates (from 40% to

90%). On the contrary, the novel framework, which generates 0.9375 *bpp* for recovery and 0.0625 *bpp* for authentication purposes (one bit for each block of 4 × 4 pixels), can obtain much more redundancy in watermarks used during the recovery process, increasing the robustness of the scheme. The novel SR-HTR framework appears to demonstrate excellent objective performance and subjective visual perception via the HVS in color image reconstruction for the highest tampering rates (from 40% to 90%).

Chia-Chen Lin et al. [37] proposed a self-embedding fragile watermarking scheme for grayscale images that uses a reference matrix as an embedding method. Each non-overlapping pixel pair is used as a coordinate in a reference matrix to embed recovery information, and each provides a 4-bit capacity as hidden space. Tampered regions are detected and restored by comparing the embedded recovery information in two LSBs of the original image with the recovery information generated from suspicious images.

In [38], a self-embedding watermarking method based on *absolute moment block truncation coding* (AMBTC) for grayscale images is presented. A checksum was introduced for accurate block authentication. They use the *optimal pixel adjustment process* (OPAP) method for embedding the recovery bits in block units in the second and third LSB and the checksum bits in the first LSB. The proposed method does not support irregular area alterations and has the limitation that if the modified region of the image is large (more than 45%), the restoration information is also removed, so that area cannot be restored.

Chin-Feng Lee et al. [39] performed a self-recovery fragile watermarking authentication scheme for grayscale images, based on two authentication methods: blockwise and pixelwise. In the first one, authentication data are generated from each block. The average block value is then used to produce recovery data; if the block's size is small, the *false positive rate* (FPR) will be reduced. For the second one, the authentication data are generated from each pixel, and the recovery data are obtained from the mean value of the block. When tampered pixels are detected, the corresponding pixel area is marked. Consequently, the FPR will be lower than blockwise detection. Since the proposed method uses blocks to detect tampered blocks, when multiple areas of the watermarked image are modified, the tampering coincidence problem significantly reduces the proposal's performance.

The majority of the previously revised methods have the principal disadvantage that the reconstructed image is low quality when significant tampering has occurred. Therefore, as mentioned above, to avoid the tampering coincidence problem, it is necessary to embed the recovery watermark on more than one occasion, as has been performed in several schemes, where two copies [10,17,23] or three copies [36] of the recovery watermark are embedded; therefore, additional chances for recovery are provided.

The main contributions of the proposed SR-HTR can be summarized as follows:

1. High quality of the watermarked image. A bit adjustment procedure is performed after the embedding process, resulting in an increased quality of a watermarking image. Several state-of-the-art methods show slightly better objective criteria values in comparison with the novel SR-HTR framework, but the detailed analysis confirms that the novel scheme appears to demonstrate acceptable objective perception quality in watermarked images as well as imperceptibility in visual subjective analysis via the HVS.

2. Better quality and robustness against the tampering coincidence problem in comparison with state-of-the-art schemes. The proposed method generates a highly compressed digest watermark, which can be embedded using three, six or nine copies of recovery watermarks, achieving higher tamper detection capability as opposed to other methods that can embed up to three times the digest image generated. The novel SR-HTR framework demonstrates the ability to reconstruct color images with high tampering rates, resulting in excellent objective and subjective performances that can be appreciated in the visual results presented below.

3. Tampering detection accuracy. During the authentication and reconstruction stage, the employed redundancy results in better tampering detection accuracy. The hierarchical tampering detection process, employed in the novel scheme, results in a decrease in false negatives during authentication and a higher manipulation detection. Better quality in the recovered image demonstrated by the proposed SR-HTR framework compared with state-of-the-art schemes is achieved via the implementation of a particular phase where the regions affected by the tampering coincidence problem should be detected. Afterward, such detected areas are processed using the inpainting method, demonstrating better quality of the recovered image in objective criteria values as well as in subjective visual perception via the HVS.

3. Designed Scheme

The designed method is divided into two stages. The first stage consists of the image protection process, which allows the insertion of multiple copies (three, six or nine) of the reconstruction watermark. This algorithm is presented in Figure 1. The second stage contains the authentication and reconstruction process, which uses a hierarchical authentication and an inpainting method to enhance the reconstruction's performance. The diagram of this stage can be observed in Figure 2.

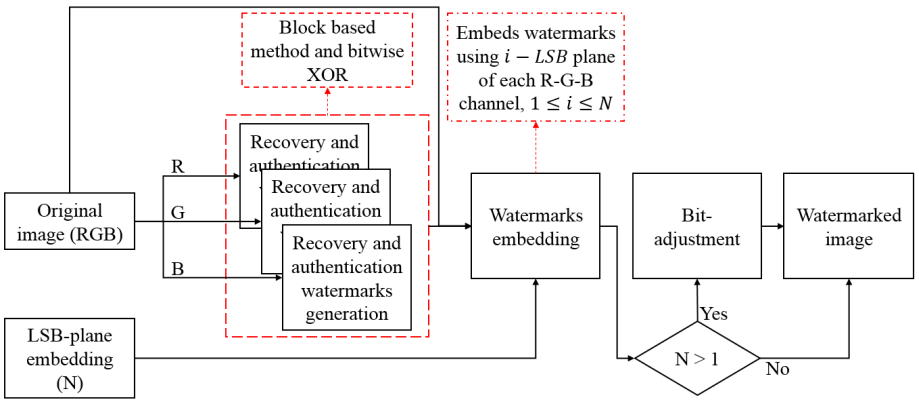


Figure 1. Protection method diagram.

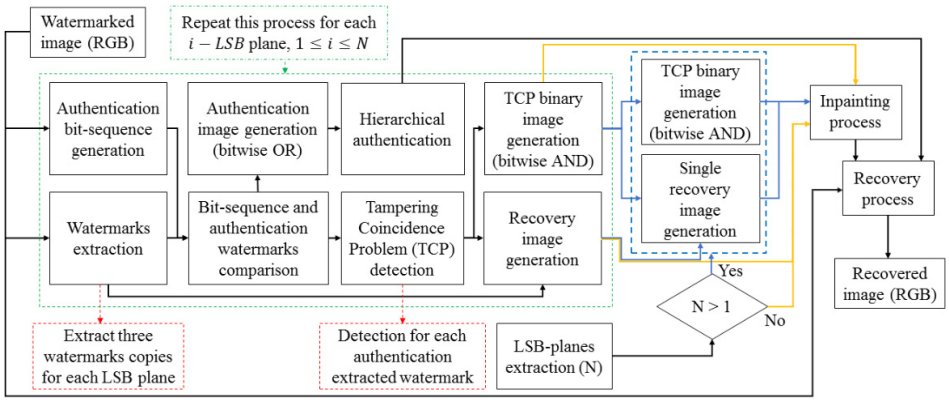


Figure 2. Reconstruction and authentication method diagram.

3.1. Image Protection

The image protection stage is described in this section. This process follows two steps: watermarking generation and insertion in the carrier image. In order to explain these steps, let us denote the original image as Ih of size $A \times B$ in the color space RGB.

3.1.1. Watermarking Generation for Reconstruction and Authentication Purposes

The original image has three channels according to its color space. Consequently, Pseudocode 1 is applied on each one of the channels, generating three watermarks for reconstruction and three for authentication. The watermarks used for reconstruction were designated as wr, wg and wb, and the ones utilized for authentication were named $autr$, $autg$ and $autb$ for the channels R, G and B, correspondingly.

Pseudocode 1 Recovery and authentication watermark generation

Input: Image to be processed **Ih**
$[A, B] = size(Ih)$
$iReference = imageResize(Ih, 0.25) \rightarrow image\ subsampling$
$iReference = bitAND(iReference, 248) \rightarrow replace\ 3\ LSBs\ by\ 0$
$recoveryW = [] \rightarrow recovery\ watermark$
$autentW = [] \rightarrow authentication\ watermark$
For $i = 1$ **to** $A/4$ **do**
For $j = 1$ **to** $B/4$ **do**
$tmpW = get5MSB\left(iReference_{i,j}\right) \rightarrow extract\ 5\ MSBs$
$recoveryW = concat(recoveryW, tmpW) \rightarrow concatenation\ process$
$aut = XOR(XOR(tmpW_1, tmpW_2), XOR(tmpW_3, tmpW_4))$
$aut = XOR(aut, tmpW_5)$
$autentW = concat(autentW, aut) \rightarrow concatenation\ process$
End for
End for

Output: Recovery watermark **recoveryW**; Authentication watermark **autentW**

Once the watermarks are obtained, a subsampled process with a factor of 0.25 is performed to significantly reduce the number of bits representing each channel of the original image. When 5 MSB of each pixel are extracted, a total number of $5AB/16$ bits are available for reconstruction watermark embedding, and $AB/16$ bits can be used for authentication watermarks. The authentication bit generation process consists of applying the bitwise XOR operation in the 5 MSBs, where a single bit is generated for each pixel block of size 4×4.

3.1.2. Watermark Embedding

The embedding process is described in Pseudocode 2, where wr, wg, wb and an authentication watermark $autr$, $autg$ or $autb$ are embedded in a selected channel of Ih. Firstly, a random permutation of the reconstruction watermarks utilizes a seed of the user key, which must be different for each processed RGB channel or bit plane.

The watermark embedding process employs the 1-LSB method to embed the watermarks for reconstruction and a single watermark for authentication. This is possible due to the size of each one of the reconstruction watermarks, which is $5AB/16$, and because a single authentication watermark is composed of $AB/16$ bits.

Finally, a bit adjustment process is applied, where each pixel in the *i,j*-th position of each RGB channel of the host image, denoted as Ihw, is compared with the pixel in the same position of Ih. This comparison has the objective to modify the intensity value of the pixels in Ihw to enhance its objective quality. This process depends on the total number of marked bit planes. For each watermarked pixel, the following equation is used:

$$Ihw_{i,j} = \begin{cases} Ih_{i,j} - 1, & \textbf{if } v \textbf{ is } \left(2^N - 1\right) \textbf{ and } \mathrm{LSB}_{N+1}\left(Ih_{i,j}\right) \textbf{ is } 1 \\ Ih_{i,j} + 1, & \textbf{if } v \textbf{ is } -\left(2^N - 1\right) \textbf{ and } \mathrm{LSB}_{N+1}\left(Ih_{i,j}\right) \textbf{ is } 0 \end{cases}, \quad (1)$$

where N represents the watermarked bit plane and $v = Ihw_{i,j} - Ih_{i,j}$ for $1 \leq i \leq A$, and $1 \leq j \leq B$. This equation can be used only for values $N > 2$.

Pseudocode 2 Watermark embedding

Input: Image (single channel) to process **Ih**; Seed **S**; Bit plane **bit Plane**; Recovery watermarks **wr**, **wg** and **wb**; Authentication watermark **aut**

$[A, B] = size(Ih)$
$rng(S)$ $\rightarrow$ *Control random number generator*
$numRand = randperm(AB/16)$ $\rightarrow$ *Random permutation of integers in range* $[1, AB/16]$
$num = 1$
For $i = 1$ **step** 4 **to** $A - 3$ **do**
For $j = 1$ **step** 4 **to** $B - 3$ **do**
$subindex = 5 * numRand[num]$
$index = subindex - 4 : subindex$ $\rightarrow$ *numbers from* $(subindex - 4)$ *to subindex*
$tmpW = concat(wr_{index}, wg_{index}, wb_{index}, aut_{num})$ $\rightarrow$ *concatenation tmpW contains* 16 *bits*
$Ihw_{i:i+3,j:j+3} = embed\left(Ih_{i:i+3,j:j+3}, bitPlane, tmpW\right)$ $\rightarrow$ *LSB embedding of tmpW in Ih by bitPlane*
$num = num + 1$
End for
End for
Equation (1), **where** $N = bit\ Plane$

Output: Watermarked image **Ihw**

3.2. Authentication and Reconstruction

This section describes the authentication and reconstruction process of a given tampered image Ihw of size $A \times B$ in RGB color space. To accomplish this reconstruction, four steps are used: extraction of the watermarks from the image, authentication of the content, post-processing to indicate the blocks affected by the tampering coincidence problem via an inpainting process to fill these regions and reconstruction of the tampered image.

3.2.1. Watermark Extraction

The extraction of the three reconstruction watermarks is performed as follows:

$$auxVal_i = \sum_{j=1}^{5} 2^{8-j} wtm_{5(i-1)+j}, \ \forall\, i \text{ s.t. } 1 \leq i \leq AB/16, \quad (2)$$

where wtm is a vector form of a watermark wr, wg or wb. The process to acquire all the watermarks is detailed in Pseudocode 3, which employs the function $vec2mat(auxVal, A/4, B/4)$ to transform $auxVal$ in a matrix of size $A/4 \times B/4$.

This pseudocode is applied to each channel image, where three reconstruction images in the RGB color space ($iRGB_R$, $iRGB_G$, $iRGB_B$) and three authentication images (aut_R, aut_G, aut_B) to authenticate each RGB channel of Ihw are obtained.

Pseudocode 3 Extraction of image watermarks

Input: Watermarked image channel **Ihw**; Seed **S**; Bit plane **bit Plane**
$[A, B] = size(Ihw)$
$rng(S) \rightarrow Control\ random\ number\ generator$
$numRand = randperm(AB/16) \rightarrow Random\ permutation\ of\ integers\ in\ range\ [1, AB/16]$
$wr = [\,] \rightarrow Empty\ list\ of\ size\ 5AB/16$
$wg = [\,] \rightarrow Empty\ list\ of\ size\ 5AB/16$
$wb = [\,] \rightarrow Empty\ list\ of\ size\ 5AB/16$
$aut = [\,] \rightarrow Empty\ list\ of\ size\ AB/16$
$num = 1$
For $i = 1$ **step** 4 **to** $A - 3$ **do**
For $j = 1$ **step** 4 **to** $B - 3$ **do**
$subindex = 5 * numRand[num]$
$index = subindex - 4 : subindex \rightarrow numbers\ from\ (subindex - 4)\ to\ subindex$
$tmpW = extract\Big(Ihw_{i:i+3,j:j+3}, bitPlane\Big) \rightarrow extract\ 16\ bits\ from\ the\ bitPlane - LSB$
$num = num + 1$
End for
End for
Equation (2), **where** $wtm = wr$
$imgWtm = vec2mat(auxVal, A/4, B/4) \rightarrow vector\ to\ matrix\ conversion$
$iRGB(:, :, 1) = imresize(imgWtm, 4)$
Equation (2), **where** $wtm = wg$
$imgWtm = vec2mat(auxVal, A/4, B/4) \rightarrow vector\ to\ matrix\ conversion$
$iRGB(:, :, 2) = imresize(imgWtm, 4)$
Equation (2), **where** $wtm = wb$
$imgWtm = vec2mat(auxVal, A/4, B/4) \rightarrow vector\ to\ matrix\ conversion$
$iRGB(:, :, 3) = imresize(imgWtm, 4)$

Output: Recovery image **iRGB**; Authentication watermark **aut**

3.2.2. Authentication

This step uses the previously described Pseudocode 1 to generate the bit sequence $autentW$ from each channel of Ihw, where $autentW_R$, $autentW_G$ and $autentW_B$ are obtained. Each bit sequence is compared with each authentication watermark aut_R, aut_G and aut_B resulting from Pseudocode 3 using the following equation:

$$autentImg_{i,j} = \begin{cases} 255, \textbf{ if } autentW_{j+(i-1)\frac{B}{4}} \textbf{ is not } aut_{j+(i-1)\frac{B}{4}} \\ 0, \textbf{otherwise} \end{cases}, \quad (3)$$

for all i and j subject to $1 \leq i \leq A/4$ and $1 \leq j \leq B/4$. Each $autentImg$ is then interpolated to size $A \times B$.

Once the previous steps have been performed, three reconstruction images ($iRGB_R$, $iRGB_G$ and $iRGB_B$) and three authentication images ($autentImg_R$, $autentImg_G$ and $autentImg_B$) are generated for each RGB channel and each LSB plane. A general authentication image is then computed by applying the bitwise OR operand to each of the authentication images, i.e., $iAutent_{i,j} = autentImg_{Ri,j} + autentImg_{Gi,j} + autentImg_{Bi,j}$, $\forall i, j$, s.t. $1 \leq i \leq A$, $1 \leq j \leq B$. A point worth mentioning is that this operation is valid only for the 1-LSB embedding method. If the 2-LSB method is utilized, six watermarks for authentication and six for reconstruction should be generated. Analogously, the 3-LSB process requires nine watermarks for each case. Therefore, $iAutent$ is obtained by applying the bitwise OR operation to the six or nine authentication images for the 2-LSB or 3-LSB method, respectively. Finally, the first level hierarchical authentication is performed on $iAutent$, improving the tamper detection recognition.

3.2.3. Post-Processing and Recovery

A binary image is generated at this stage, representing the blocks affected by the tampering coincidence problem in each reconstruction image generated by Pseudocode 3 ($iRGB_R$, $iRGB_G$, $iRGB_B$). Each authentication image generated in Section 3.2.2 ($autentImg_R$, $autentImg_G$, $autentImg_B$) is used for this process.

Pseudocode 4 presents the image generation process as follows:

$$autent_{numrand[j+(i-1)B/4]} = \begin{cases} 255, & \textbf{if } autentImg_{i,j} \textbf{ is } 255 \\ 0, & \textbf{otherwise} \end{cases}, \quad (4)$$

for all i and j subject to $1 \leq i \leq A/4$, and $1 \leq j \leq B/4$. Finally, the generated image is interpolated to $A \times B$ size. Pseudocode 4 presents the processes, resulting in three images being obtained, denoted as TCP_R, TCP_G and TCP_B.

If a large amount of the original image information is altered, the information used for reconstruction could be overwritten, although it is redundant. Therefore, maps to point to the tampering coincidence problem are computed using binary images (Pseudocode 4). Then, an AND operand between each one of these maps' bits is applied, resulting in a binary image denoted as $iTCP$, i.e., $iTCP_{i,j} = TCP_{Ri,j} * TCP_{Gi,j} * TCP_{Bi,j}$, $\forall\, i,j$, s.t. $1 \leq i \leq A$, $1 \leq j \leq B$. This image marks the regions affected by the tampering coincidence problem.

Pseudocode 4 Detection of tampering coincidence problem

Input: Authentication image **autent Img**; Seed **S**
$[A, B] = size(autentImg)$
$rng(S)$ $\rightarrow$ *Control random number generator*
$numRand = randperm(AB/16)$ $\rightarrow$ *Random permutation of integers in range* $[1, AB/16]$
$autentImg = imresize(autentImg, 0.25)$
Equation (4)
$TCP = vec2mat(autent, A/4, B/4)$ $\rightarrow$ *vector to matrix conversion*
$TCP = imresize(TCP, 4)$

Output: Tampering coincidence problem image **TCP**

Subsequently, Equation (5) is applied to the reconstruction images obtained by Pseudocode 3 and their corresponding binary images given by the Pseudocode 4, resulting in a single reconstruction image named iR :

$$iR_{i,j} = \frac{\sum_{\alpha=1}^{n}\left(1 - TCP_{\alpha_{i,j}}\right) Iw_{\alpha_{i,j}}}{\sum_{\alpha=1}^{n}\left(1 - TCP_{\alpha_{i,j}}\right)}, \quad \forall\, i,j \,:\, 1 \leq i \leq A,\ 1 \leq j \leq B, \quad (5)$$

where α represents the i-th processed copy, Iw represents a recovery image, TCP represents a binary image, which indicates the regions affected by the tampering coincidence problem of Iw, where the values "1" indicate that the i, j-th position is authentic and a value of "0" indicates that this position was affected by this problem. Implementation of Equation (5) is given for $TCP = [TCP_R, TCP_G, TCP_B]$ and $Iw = [iRGB_R, iRGB_G, iRGB_B]$.

The previous sequence is valid when the 1-LSB method is selected for embedding. If the scheme is presented in their variants 2-LSB or 3-LSB, further copies of the watermarks should be computed from the bit panels, and the reconstruction method requires single instances of iR and $iTCP$. Consequently, the following equations for the N-LSB method must be applied:

$$iTCP = \begin{cases} iTCP_1, & \textbf{if } N \textbf{ is } 1 \\ iTCP_1 * iTCP_2, & \textbf{if } N \textbf{ is } 2 \\ iTCP_1 * iTCP_2 * iTCP_3, & \textbf{if } N \textbf{ is } 3 \end{cases}, \quad (6)$$

$$iR = \begin{cases} iR_1, & \textbf{if } N \textbf{ is } 1 \\ \text{Equation (5), where } Iw = [iR_1, iR_2, iR_2] \text{ and } TCP = [iTCP_1, iTCP_2, iTCP_2], & \textbf{if } N \textbf{ is } 2 \\ \text{Equation (5), where } Iw = [iR_1, iR_2, iR_3] \text{ and } TCP = [iTCP_1, iTCP_2, iTCP_3], & \textbf{if } N \textbf{ is } 3 \end{cases}, \quad (7)$$

Finally, Pseudocode 5 performs an inpainting method for the given iR and $iTCP$ images. The output image is named as $iRecovery$. The inpainting method divides the iR and $iTCP$ images into overlapping blocks of 3×3 pixels, and the following equations are used for each block:

$$iR_{i,j} = \frac{\sum_{a=1}^{3}\sum_{b=1}^{3}(wIo. * (1 - wTCP))_{a,b}}{\sum_{a=1}^{3}\sum_{b=1}^{3}(1 - wTCP_{a,b})}, \textbf{if } wTCP_{2,2} \textbf{ is } 1 \textbf{ and } \sum_{i=1}^{3}\sum_{j=1}^{3}(1 - wTCP_{i,j}) > 1, \quad (8)$$

$$iTCP_{i,j} = 0, \textbf{ if } wTCP_{2,2} \textbf{ is } 1 \textbf{ and } \sum_{i=1}^{3}\sum_{j=1}^{3}(1 - wTCP_{i,j}) > 1, \quad (9)$$

where $wTCP$ represents a block of the $iTCP$ image, which is subject to $wTCP = iTCP_{i-1:i+1,j-1:j+1}$, and wIo is a block of the iR image, given $wIo = iR_{i-1:i+1,j-1:j+1}$, for all i, j, such that $2 \leq i \leq A + 1, 2 \leq j \leq B + 1$.

Finally, the tampered zones of the Ihw image can be reconstructed by means of the following equation:

$$Ihw_{i,j} = Ihw_{i,j}(1 - iAutent_{i,j}) + (iRecovery_{i,j})(iAutent_{i,j}), \ 1 \leq i \leq A, \ 1 \leq j \leq B. \quad (10)$$

Pseudocode 5 Inpainting application

Input: Image to be processed **iR**; Binary image **iTCP**
$[A, B] = size(iR)$
While $\sum_{i=1}^{A}\sum_{j=1}^{B} iTCP_{i,j} != 0$ **do**
$iR = \begin{bmatrix} iR_{1,1} & iR_{1,:} & iR_{1,B} \\ iR_{:,1} & iR_{:,:} & iR_{:,B} \\ iR_{A,1} & iR_{A,:} & iR_{A,B} \end{bmatrix} \rightarrow Matrix\ of\ size\ A+2, B+2$
$iTCP = \begin{bmatrix} iTCP_{1,1} & iTCP_{1,:} & iTCP_{1,B} \\ iTCP_{:,1} & iTCP_{:,:} & iTCP_{:,B} \\ iTCP_{A,1} & iTCP_{A,:} & iTCP_{A,B} \end{bmatrix} \rightarrow Matrix\ of\ size\ A+2, B+2$
$iTCP = iTCP > 127$
Equations (8) and (9)
$iTCP = iTCP * 255$
$iR = iR_{2:A-1,2:B-1} \rightarrow Matrix\ of\ size\ A, B$
$iTCP = iTCP_{2:A-1,2:B-1} \rightarrow Matrix\ of\ size\ A, B$
End While
$iRecovery = iR$

Output: Image after inpainting process **iRecovery**

3.3. Implementation of the Algorithms

The proposed SR-HTR method allows the insertion of N copies of the three reconstruction watermarks for the N-LSB method limited by $1 \leq N \leq 3$. However, the algorithm implementation has to be changed depending on the parameter N, as can be observed in Pseudocodes 6 and 7.

Pseudocode 6 Image protection

Input: Image (RGB) to be protected **Ih**; LSB plane **N**
$wr,\ autr\ =$ Pseudocode 1 $(Ih(:,:,1))$
$wg,\ autg\ =$ Pseudocode 1 $(Ih(:,:,2))$
$wb,\ autb\ =$ Pseudocode 1 $(Ih(:,:,3))$
$Ihw\ =\ Ih$
For $i\ = 1$ **step** 1 **to** N **do**
$Ihw(:,:,1) =$ Pseudocode 2 $(Ihw(:,:,1), 10+i, i, wr, wg, wb, autr)$
$Ihw(:,:,2) =$ Pseudocode 2 $(Ihw(:,:,2), 20+i, i, wr, wg, wb, autg)$
$Ihw(:,:,3) =$ Pseudocode 2 $(Ihw(:,:,3), 30+i, i, wr, wg, wb, autb)$
End For

Output: Watermarked image **Ihw**

Pseudocode 7 Image authentication and recovery

Input: Suspicious image (RGB) **Ihw**; LSB plane **N**
For $i\ = 1$ **step** 1 **to** N **do**
$iRGB_{R_i},\ aut_{Ri}\ =$ Pseudocode 3 $(Ihw(:,:,1), 10+i, i)$
$iRGB_{G_i},\ aut_{Gi}\ =$ Pseudocode 3 $(Ihw(:,:,2), 20+i, i)$
$iRGB_{B_i},\ aut_{Bi}\ =$ Pseudocode 3 $(Ihw(:,:,3), 30+i, i)$
End For
$\sim,\ autentW_R\ =$ Pseudocode 1 $(Ihw(:,:,1))$
$\sim,\ autentW_G\ =$ Pseudocode 1 $(Ihw(:,:,2))$
$\sim,\ autentW_B\ =$ Pseudocode 1 $(Ihw(:,:,3))$
$iAutent\ =\ zeros(A,B)$
For $i\ = 1$ **step** 1 **to** N **do**
Equation (3), **where** $autentW = autentW_R$ **and** $aut = aut_{Ri}$
$autentImg_{R_i} = imresize(autentImg, 4)$
Equation (3), **where** $autentW = autentW_G$ **and** $aut = aut_{Gi}$
$autentImg_{G_i} = imresize(autentImg, 4)$
Equation (3), **where** $autentW = autentW_B$ **and** $aut = aut_{Bi}$
$autentImg_{B_i} = imresize(autentImg, 4)$
$A_i\ = autentImg_{R_i} + autentImg_{G_i} + autentImg_{B_i} \rightarrow$ OR operation
$iAutent\ = iAutent\ +\ A_i \rightarrow$ OR operation
End For
$iAutent\ = hierarchical\_authentication(iAutent)$
For $i\ = 1$ **step** 1 **to** N **do**
$TCP_{R_I} =$ Pseudocode 4 $(autentImg_{R_i},\ 10+i)$
$TCP_{G_I} =$ Pseudocode 4 $(autentImg_{G_i},\ 20+i)$
$TCP_{B_I} =$ Pseudocode 4 $(autentImg_{B_i},\ 30+i)$
Equation (5), **where** $Iw = [iRGB_{R_i}, iRGB_{G_i}, iRGB_{B_i}]$ **and** $TCP = [TCP_{R_I}, TCP_{G_I}, TCP_{B_I}]$
$iTCP_i\ = TCP_{R_I} * TCP_{G_I} * TCP_{B_I} \rightarrow$ AND operation
End For
Equations (6) and (7)
$iRecovery\ =$ Pseudocode 5 $(iR,\ iTCP)$
Equation (10)

Output: Restored image **Ihw**

4. Experimental Setup

Images of sizes 512×768 and 768×512 from the Kodak database [40], which consists of 24 images, were used for experimentation. These images are labeled as Kodak-n, where n is the image identifier. Some of these images are shown in Figure 3.

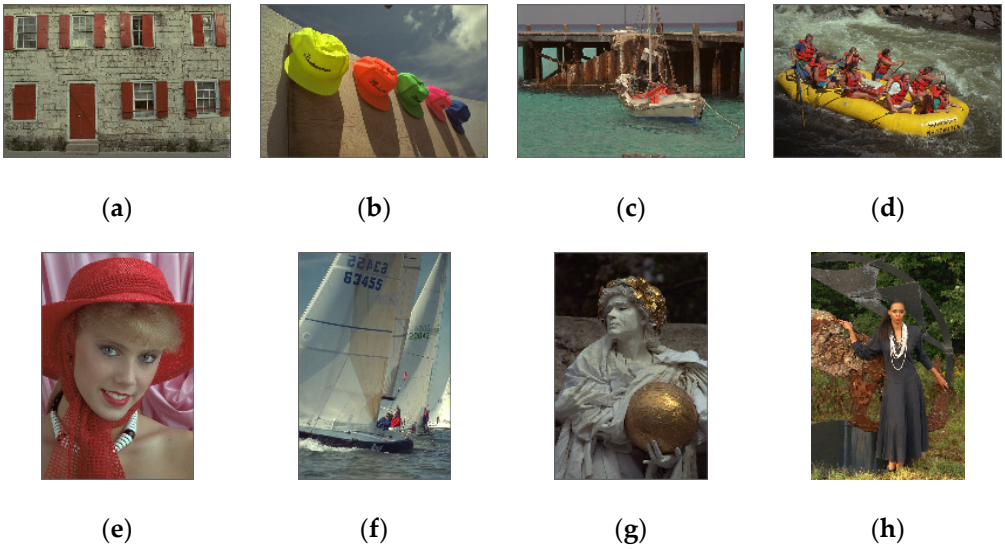


Figure 3. Images employed from Kodak database [40], (**a**) Kodak-1, (**b**) Kodak-3, (**c**) Kodak-11, (**d**) Kodak-14, (**e**) Kodak-4, (**f**) Kodak-10, (**g**) Kodak-17, (**h**) Kodak-18.

The performance criteria to evaluate the quality of watermarked and recovered images obtained by the proposed framework are: PSNR, SSIM and PSNR-HVS-M [35]. The PSNR metric is defined as follows:

$$PSNR = 10\, log_{10} \frac{MAX^2}{MSE}, \tag{11}$$

$$MSE = \frac{1}{XY} \sum_{i=0}^{X-1} \sum_{j=0}^{Y-1} \left[I(i,j) - I'(i,j) \right]^2, \tag{12}$$

where I represents the original image, $I\prime$ corresponds to the modified image, the arguments i, j are used for the pixel position and X, Y are the number of rows and columns, correspondingly. The SSIM metric is computed using the following equation:

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x{}^2 + \mu_y{}^2 + C_1)(\sigma_x{}^2 + \sigma_y{}^2 + C_2)}, \tag{13}$$

where μ and σ denote the mean and variance of images x and y, σ_{xy} is the covariance between x and y, constants C_1, C_2 are: $C_1 = (0.01L)^2$, $C_2 = (0.03L)^2$, and $L = 255$ [41].

Furthermore, *Precision* and *Recall* metrics were utilized to measure the alteration detection performance of the proposed method. These metrics are based on the numbers of true positives (TP), false positives (FP) and false negatives (FN) among all pixels:

$$Precision = \frac{TP}{TP + FP}, \tag{14}$$

$$Recall = \frac{TP}{TP + FN}. \tag{15}$$

5. Analysis of 1-LSB, 2-LSB and 3-LSB Schemes in Embedding Stage

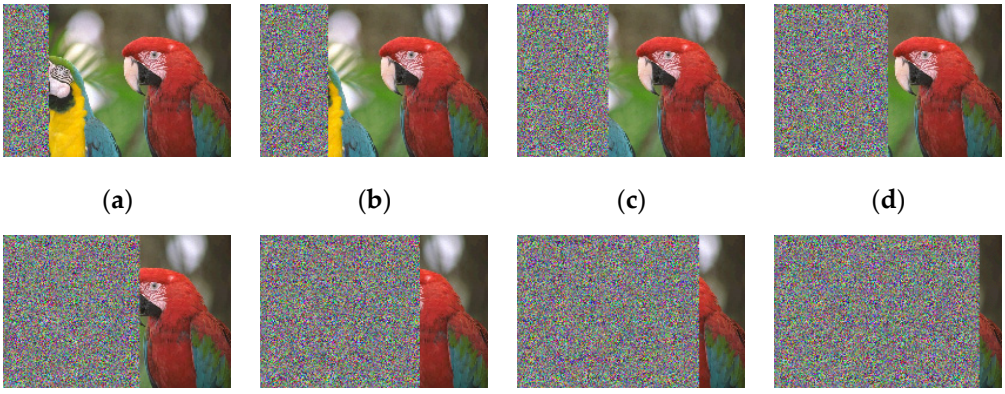
A comparison between the different LSB variants of the designed framework was performed. The watermarked images were compared with the original ones, and the results are reported in Table 1, which shows the average values of the objective quality measures PSNR, SSIM and PSNR-HVS-M for each variant. Furthermore, the comparison was also performed using the bit adjustment for 2-LSB and 3-LSB methods. It can be observed that a minimal enhancement is generally achieved using this adjustment.

Table 1. Least-significant bit (LSB) embedding analysis in terms of the objective quality.

	No Bit Adjustment			With Bit Adjustment	
Embedding	**1-LSB**	**2-LSB**	**3-LSB**	**2-LSB**	**3-LSB**
PSNR	51.17	43.89	37.51	44.33	37.69
SSIM	0.9966	0.9824	0.9314	0.9843	0.9340
PSNR-HVS	59.21	49.89	41.80	49.69	41.92

During the imperceptibility evaluation of the three variants (1-LSB, 2-LSB or 3-LSB) shown in Table 1, we can observe that the 1-LSB and the 3-LSB variants have the best and the worst performance, respectively, in terms of objective quality evaluation. However, a point worth mentioning is that for the worst case (3-LSB), the embedding of nine copies of the watermark does not produce any recognizable visual modification in the watermarked image.

To measure the authentication and reconstruction process's performance, experiments changing the tampering rate from 10% to 90% in the image were carried out, adding pseudo-random noise to the image, as can be observed in Figure 4.



(**a**) (**b**) (**c**) (**d**)

(**e**) (**f**) (**g**) (**h**)

Figure 4. Tampered images for Kodak-23, (**a**) 20%, (**b**) 30%, (**c**) 40%, (**d**) 50%, (**e**) 60%, (**f**) 70%, (**g**) 80%, (**h**) 90%.

The authentication stage evaluation is reported in Table 2, where it can be observed that the 1-LSB method maintains the best results in *Precision*. In terms of *Recall*, the proposed variants' performance increases when this measure is close to one. The variants 2-LSB and 3-LSB have the same high Recall value, even though the 3-LSB variant embeds more copies. Additionally, the probability that the 1-LSB variant extracts copies incorrectly is higher than the other variants, generating more errors during the authentication process. Another point worth mentioning is that the 2-LSB variant achieves Recall's best performance, avoiding the tampering coincidence problem, embedding only six copies of the recovery watermark.

Table 2. *Precision* and *Recall* obtained from the authentication process.

	Embedding	**10%**	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
Recall	1-LSB	0.9033	0.9211	0.9369	0.9345	0.9356	0.9386	0.9404	0.9389	0.9384
	2-LSB	1	1	1	1	1	1	1	1	1
	3-LSB	1	1	1	1	1	1	1	1	0.9999
Precision	1-LSB	0.9466	0.9733	0.9831	0.988	0.9943	0.9913	0.9922	0.9932	0.9944
	2-LSB	0.9157	0.9585	0.9718	0.9799	0.9885	0.9849	0.9877	0.9892	0.9910
	3-LSB	0.8893	0.9448	0.9637	0.9744	0.9843	0.9819	0.9855	0.9874	0.9900

Finally, the average values of the quality measures PSNR, SSIM and PSNR-HVS-M for the reconstructed images compared with the original ones are shown in Table 3. A significant increase in the values with the 2-LSB method can be recognized. Nevertheless,

although the 3-LSB implementation also showed a sharp increment compared with 1-LSB, its results are slightly lower than the 2-LBS method. On the contrary, the PSNR-HSV-M values of the 3-LSB method are slightly higher than the result from 2-LSB. However, the 2-LSB results are still acceptable, and, in general, their PSNR and SSIM values are superior to the other implementations in this study. Consequently, the 2-LSB variation was utilized for the proposed SR-HTR method considering the results given in Tables 1–3.

Table 3. PSNR, SSIM and PSNR-HVS-M from the reconstruction process.

	Embedding	10%	20%	30%	40%	50%	60%	70%	80%	90%
PSNR	1-LSB	27.81	25.02	23.50	21.79	20.30	19.01	17.57	15.79	13.286
	2-LSB	35.80	32.71	30.63	29.25	28.13	27.06	26.00	24.82	23.07
	3-LSB	35.52	32.48	30.43	29.08	28.03	27.06	26.22	25.27	23.68
SSIM	1-LSB	0.953	0.903	0.851	0.787	0.713	0.629	0.533	0.410	0.250
	2-LSB	0.967	0.935	0.901	0.867	0.832	0.793	0.750	0.696	0.615
	3-LSB	0.962	0.925	0.888	0.851	0.813	0.772	0.730	0.680	0.604
PSNR-HVS-M	1-LSB	26.19	22.83	20.91	18.94	17.04	15.26	13.54	11.49	8.78
	2-LSB	33.29	30.19	28.14	26.69	25.57	24.37	23.02	21.32	18.87
	3-LSB	33.25	30.18	28.15	26.73	25.69	24.71	23.76	22.35	19.90

6. Experimental Results and Discussion

Since the 2-LSB method maintains a balance between the quality of the marking, authentication and reconstruction processes, it was selected to insert the watermarks. In this section, the proposed method's performance with the 2-LSB variation is detailed and compared with other state-of-the-art methods.

6.1. Watermarked Image Quality

Table 4 shows the results after the watermark embedding and the variation with and without the bit adjustment stage. It can be observed that the bit adjustment markedly raised the PSNR and SSIM values. Nonetheless, the best PSNR-HSV-M values fluctuated between both implementations.

Table 4. Objective quality metrics PSNR, SSIM and PSNR-HVS-M for watermarked images.

	With Bit Adjustment			Without Bit Adjustment		
	PSNR	SSIM	PSNR-HVS	PSNR	SSIM	PSNR-HVS
Kodak-1	44.36	0.9923	52.49	43.93	0.9914	53.30
Kodak-2	43.87	0.9787	49.02	43.44	0.9756	50.17
Kodak-3	44.26	0.9762	48.02	43.81	0.9732	47.96
Kodak-4	44.30	0.9819	49.73	43.90	0.9798	50.05
Kodak-5	44.20	0.9922	50.89	43.83	0.9914	52.12
Kodak-6	44.41	0.9874	49.10	43.92	0.9859	48.82
Kodak-7	44.31	0.9817	48.99	43.92	0.9797	49.10
Kodak-8	44.41	0.9934	52.16	43.94	0.9925	52.50
Kodak-9	44.47	0.9790	49.10	44.02	0.9764	48.94
Kodak-10	44.45	0.9806	49.90	43.97	0.9783	49.57
Kodak-11	44.34	0.9854	50.99	43.84	0.9834	50.80
Kodak-12	44.34	0.9788	49.47	43.82	0.9761	48.84
Kodak-13	44.30	0.9948	52.49	43.88	0.9941	53.46
Kodak-14	44.27	0.9896	50.76	43.89	0.9884	51.67
Kodak-15	44.34	0.9800	48.10	43.92	0.9777	47.74
Kodak-16	44.32	0.9825	49.82	43.89	0.9805	49.87
Kodak-17	44.17	0.9819	48.40	43.84	0.9800	48.82
Kodak-18	44.14	0.9879	50.11	43.77	0.9865	51.20
Kodak-19	44.44	0.9842	51.12	43.96	0.9823	51.03
Kodak-20	44.89	0.9845	44.99	44.27	0.9832	44.32
Kodak-21	44.32	0.9828	49.16	43.91	0.9808	49.27
Kodak-22	44.41	0.9853	50.35	43.95	0.9834	50.38
Kodak-23	44.35	0.9758	48.43	43.94	0.9731	48.39
Kodak-24	44.27	0.9876	49.16	43.84	0.9862	49.20

6.2. Analysis of Tampering Detection

In order to measure the performance of the proposed method in change detection, different modifications were applied to the test watermarked images. Afterward, the authentication process was executed using the six authentication watermarks and the hierarchical authentication method to discard most of the detection errors, specifically the false negatives.

The tampering detection process evaluation consists of two alteration schemes: the first one was used to estimate the ability to detect alteration rates between 10% and 90% by adding a regular square area of pseudo-random noise into the watermarked image, as displayed in Figure 4. In the second scheme, the alterations were performed by modifying one or multiple irregular areas in the watermarked image using Adobe Photoshop software, maintaining the structure and original nature of the image and avoiding significant falsity in the alteration. Figure 5 illustrates the watermarked images with irregular alterations of Kodak-1 in 46.41%, Kodak-3 in 34.34%, Kodak-11 in 25.06% and Kodak-14 in 40.08% of the entire image.

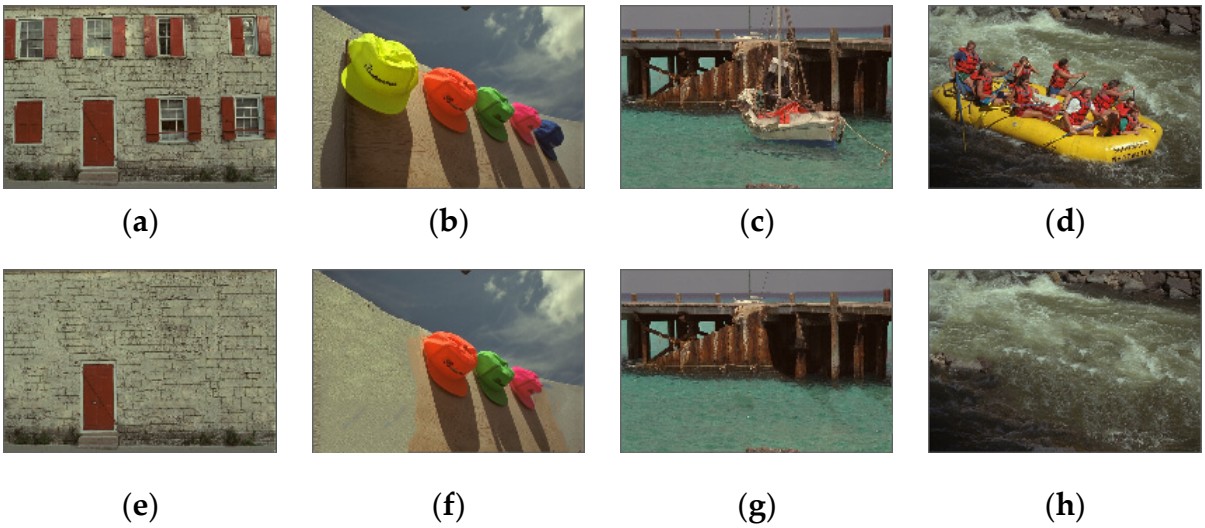


Figure 5. Multiple and irregular alterations, (**a**) Kodak-1 original, (**b**) Kodak-3 original, (**c**) Kodak-11 original, (**d**) Kodak-14 original, (**e**) Kodak-1 altered, (**f**) Kodak-3 altered, (**g**) Kodak-11 altered, (**h**) Kodak-14 altered.

The detection's *Precision* results for the changes using the first scheme are shown in Table 5. It can be noticed that the *Precision* performance is enhanced as the alteration rate is increased. The Recall metric resulted in values of 1.0 for all the alteration rates between 10% and 80% and 0.9999 for a change of 90% of the image. This is due to the hierarchical method of authentication.

The evaluation of the second scheme, which uses multiple and irregular alterations in the images, is displayed in Figure 6, where images in Figure 6a–d represent the ground truth of the alterations, and images in Figure 6e–h are the results of the change detection obtained by SR-HTR. The pairs of (*Precision*, *Recall*) values are: (0.9190, 0.9995) for Kodak-1, (0.9416, 1.0) for Kodak-3, (0.9214, 1.0) for Kodak-11 and (0.9382, 0.9998) for Kodak-14.

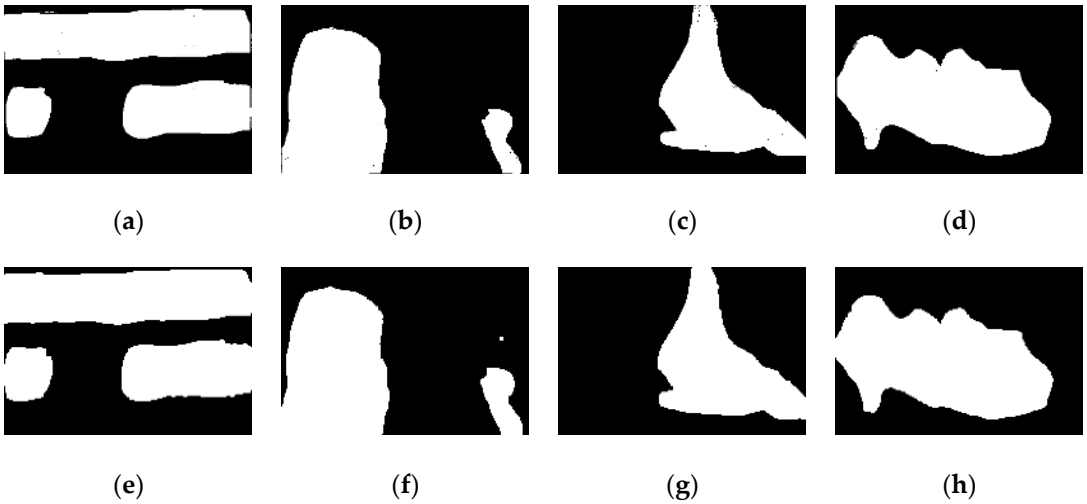


Figure 6. Multiple and irregular alterations, (**a**) Kodak-1 ground truth, (**b**) Kodak-3 ground truth, (**c**) Kodak-11 ground truth, (**d**) Kodak-14 ground truth, (**e**) detection for Kodak-1, (**f**) detection for Kodak-3, (**g**) detection for Kodak-11, (**h**) detection for Kodak-14.

Table 5. *Precision* values for the detection of different alteration rates between 10% and 90%.

	10%	20%	30%	40%	50%	60%	70%	80%	90%
Kodak-1	0.9170	0.9629	0.9748	0.9840	0.9898	0.9850	0.9890	0.9903	0.9930
Kodak-2	0.9167	0.9631	0.9748	0.9844	0.9898	0.9851	0.9894	0.9904	0.9928
Kodak-3	0.9177	0.9625	0.9747	0.9840	0.9898	0.9850	0.9892	0.9904	0.9929
Kodak-4	0.9107	0.9454	0.9630	0.9673	0.9846	0.9845	0.9835	0.9857	0.9854
Kodak-5	0.9170	0.9625	0.9746	0.9841	0.9898	0.9852	0.9892	0.9903	0.9930
Kodak-6	0.9170	0.9629	0.9748	0.9841	0.9898	0.9851	0.9891	0.9906	0.9929
Kodak-7	0.9170	0.9627	0.9747	0.9840	0.9898	0.9850	0.9890	0.9906	0.9929
Kodak-8	0.9180	0.9629	0.9747	0.9840	0.9898	0.9852	0.9891	0.9904	0.9929
Kodak-9	0.9107	0.9454	0.9629	0.9676	0.9846	0.9842	0.9835	0.9856	0.9851
Kodak-10	0.9107	0.9450	0.9630	0.9670	0.9846	0.9844	0.9836	0.9859	0.9853
Kodak-11	0.9177	0.9627	0.9746	0.9841	0.9898	0.9850	0.9890	0.9903	0.9929
Kodak-12	0.9177	0.9627	0.9748	0.9841	0.9898	0.9851	0.9890	0.9904	0.9929
Kodak-13	0.9170	0.9629	0.9748	0.9841	0.9898	0.9852	0.9891	0.9903	0.9929
Kodak-14	0.9167	0.9640	0.9746	0.9843	0.9898	0.9853	0.9891	0.9903	0.9928
Kodak-15	0.9167	0.9627	0.9747	0.9842	0.9898	0.9851	0.9891	0.9905	0.9930
Kodak-16	0.9180	0.9629	0.9747	0.9843	0.9898	0.9851	0.9890	0.9904	0.9928
Kodak-17	0.9107	0.9452	0.9629	0.9674	0.9846	0.9840	0.9838	0.9857	0.9853
Kodak-18	0.9117	0.9455	0.9626	0.9671	0.9846	0.9841	0.9838	0.9857	0.9852
Kodak-19	0.9121	0.9446	0.9626	0.9678	0.9846	0.9841	0.9836	0.9857	0.9851
Kodak-20	0.9167	0.9634	0.9747	0.9842	0.9898	0.9850	0.9890	0.9903	0.9929
Kodak-21	0.9174	0.9633	0.9748	0.9842	0.9898	0.9850	0.9890	0.9903	0.9929
Kodak-22	0.9174	0.9625	0.9748	0.9840	0.9898	0.9850	0.9894	0.9905	0.9929
Kodak-23	0.9174	0.9627	0.9747	0.9841	0.9898	0.9851	0.9892	0.9904	0.9928
Kodak-24	0.9177	0.9629	0.9747	0.9840	0.9898	0.9851	0.9890	0.9904	0.9931

6.3. Evaluation of the Reconstruction under Different Tampering Rates

In this section, the results for the evaluation of the reconstruction process are reported. The reconstruction of the test images was evaluated using both alteration schemes. For the first scheme, which is illustrated in Figure 4, the six versions of the reconstruction watermark are utilized for reconstruction. The inpainting method was performed to regenerate the information affected by the tampering coincidence problem.

Figure 7 shows the reconstructed images for Kodak-23. As can be noticed, the reconstruction capability for each alteration rate markedly dropped according to the increment in the alteration rate. Nevertheless, the original content of the image can be clearly distinguished.

(**a**) (**b**) (**c**)

(**d**) (**e**) (**f**)

(**g**) (**h**)

Figure 7. Quality measures values (PSNR/SSIM/PSNR-HVS-M) for Kodak-23 reconstruction given different alteration rates: (**a**) 20%, 37.91/0.9762/34.69; (**b**) 30%, 31.99/0.9552/28.82; (**c**) 40%, 31.26/0.9401/27.96; (**d**) 50%, 30.66/0.9250/27.47; (**e**) 60%, 29.96/0.9079/26.68; (**f**) 70%, 28.21/0.8739/24.69; (**g**) 80%, 27.25/0.8418/23.28; (**h**) 90%, 25.21/0.7854/20.47.

Additionally, it can be observed that a granulated effect is obtained according to the alteration rate. The inpainting process gives this effect due to the replenishment of affected areas by the tampering coincidence problem, with neighboring pixels' intensity values detected as authentic ones.

Finally, this process was executed on all the test images using the first alteration scheme. Figures 8–10 report the PSNR, SSIM and PSNR-HVS-M results after the reconstruction process, correspondingly. These graphics show an average decrease of 12.72 dB for PSNR and 14.41 dB for PSNR-HSV-M in reconstructing the modification rates between 10% and 90% of the image and an average reduction of 0.3516 for SSIM values. The results can be considered acceptable given the high alteration rate to which the images are subjected. A point worth mentioning is that other works usually test their methods with an upper boundary of 50% of modification since they do not consider the tampering coincidence problem consequences in their entirety.

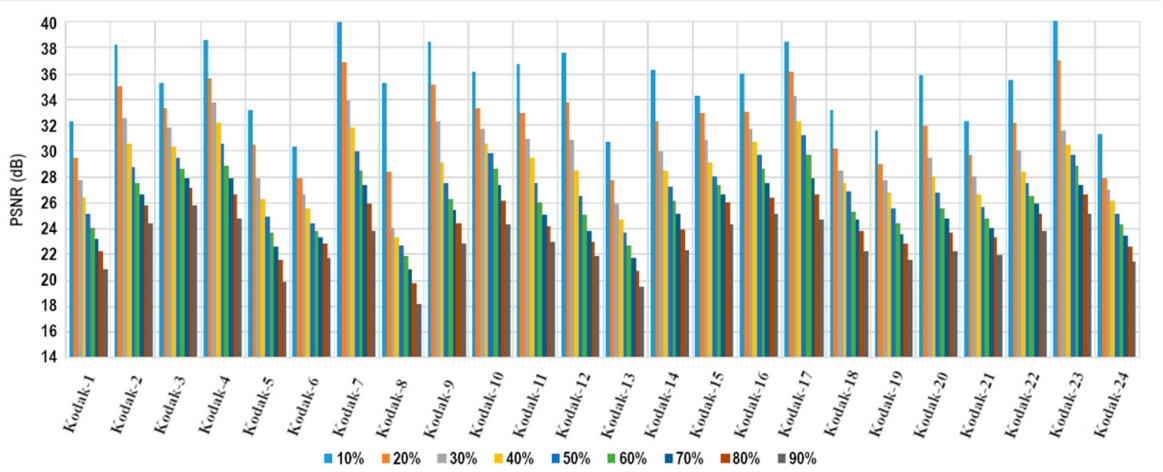


Figure 8. PSNR of the reconstructed images for different alteration rates.

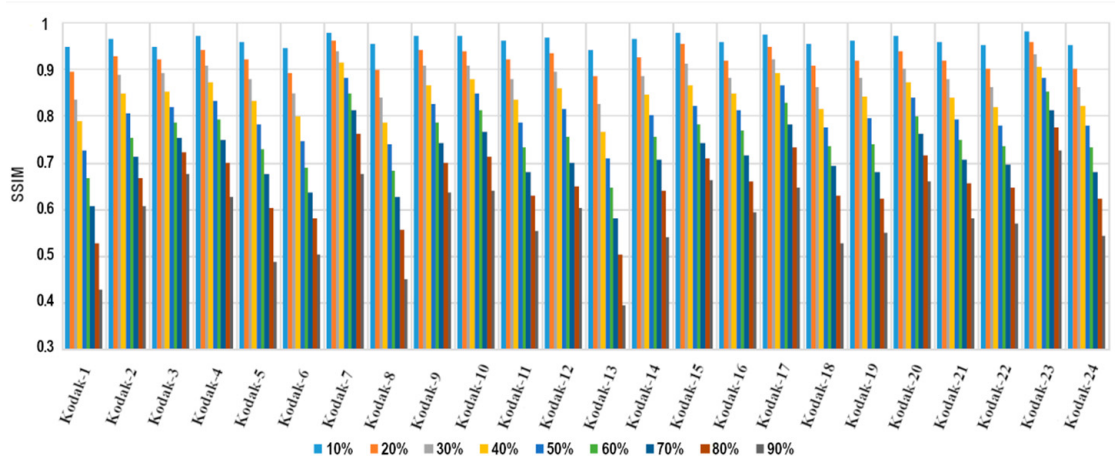


Figure 9. SSIM of the reconstructed images for different alteration rates.

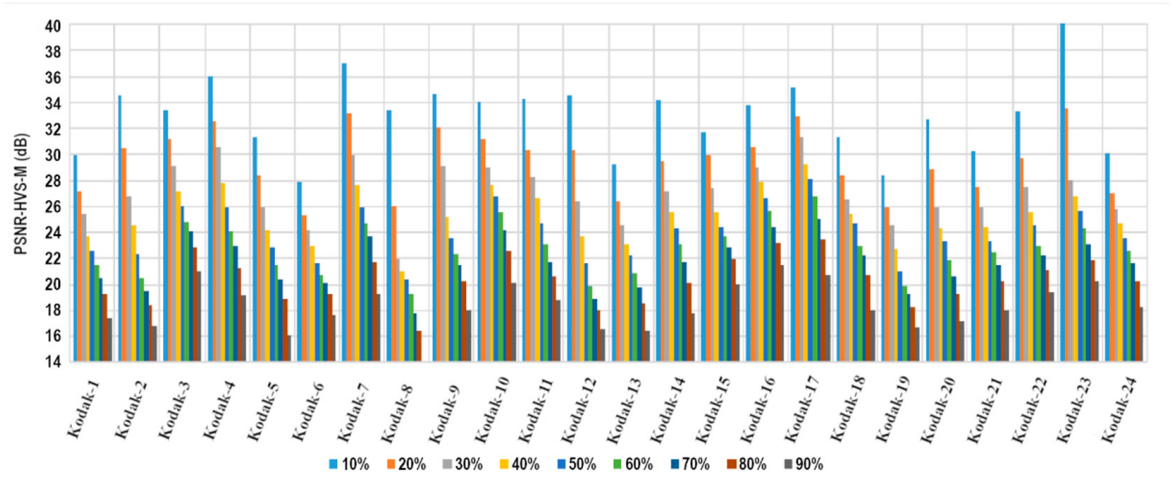


Figure 10. PSNR-HVS-M of the reconstructed images for different alteration rates.

6.4. Evaluation of the Image Reconstruction under Multiple and Irregular Attacks

The results of the reconstructed images that were modified using the second alteration scheme, given in Figure 5, are presented in this section. Figure 11 shows a satisfactory

visual quality of the reconstructed images, which display the original content substituted with other information.

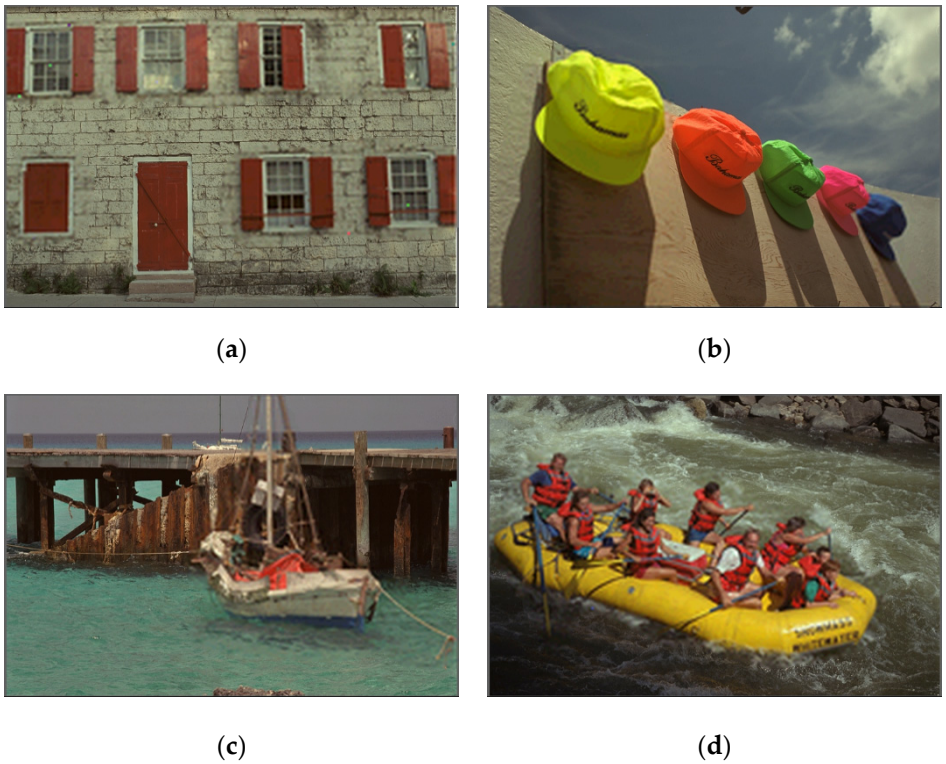


Figure 11. Reconstructed images from multiple and irregular alterations (PSNR/SSIM/PSNR-HVS-M), (**a**) Kodak-1 (24.53/0.7417/21.94), (**b**) Kodak-3 (31.82/0.8959/29.27), (**c**) Kodak-11 (28.12/0.8670/26.08), (**d**) Kodak-14 (26.38/0.8263/23.41).

6.5. Comparison with State-of-the-Art Schemes

The proposed SR-HTR method was compared in terms of performance with other state-of-the-art methods. As previously mentioned, the design of SR-HTR aims to achieve high performance when there is a high rate of modification in a watermarked image. This is accomplished by the insertion of redundancy in the authentication and reconstruction bits. Therefore, the comparison is performed in terms of objective quality in the following areas: watermarked image visualization, change detection rates and reconstruction image visualization.

The first evaluation consists of the comparison between the watermarked image and the original one. Table 6 shows the average values of PSNR, SSIM and PSNR-HVS-M for the set of test images employed.

Table 6. Quality comparison using PSNR, SSIM and PSNR-HVS-M between watermarked images and original ones.

	PSNR (dB)	SSIM	PSNR-HVS (dB)
SR-HTR	44.33	0.9843	49.69
Molina [36]	44.32	0.9845	49.36
Fan [23]	44.08	0.9832	50.09
Singh [2]	37.85	0.9364	42.04
Tai [24]	44.08	0.9832	50.07
Tong [10]	37.85	0.9363	42.05

The novel SR-HTR method and Molina [36] present the best quality results in PSNR and SSIM. This is due to the bit adjustment stage implemented in 2-LSB for both methods.

For the PSNR-HVS-M metric, the highest values belong to [23,24], followed by the proposed method, [36], and then by [2,10]. A point worth mentioning is that the bit adjustment of SR-HTR and [36] negatively affects the results of PSNR-HSV-M due to the characteristics of this metric, as described in Tables 1 and 4. Furthermore, it is essential to notice that the methods [23,24,36] employ the 2-LSB method, and procedures [2,10] use 3-LSB insertion. Additionally, methods [23,24] do not exploit the total insertion capacity to use enough redundant information in the watermarks, leading to lower performance in the reconstruction stage.

The second evaluation is related to change detection. For this test, *Precision* and *Recall* metrics were employed to compare the performance of the alteration schemes previously described.

The *Precision* results of this evaluation using the first alteration scheme are shown in Table 7. The best results belong to [2,10]. Compared with the other methods, these methods possess an authentication scheme that generates multiple authentication bits for each block of $n \times n$ pixels. Consequently, there is a higher likelihood to detect an alteration in a block because the number of bits to be compared increases. However, the proposed method was designed to generate a single authentication bit for each block of 4×4 pixels, and this significantly reduces the detection capability.

Table 7. Comparison of Precision for different alteration rates between 10% and 90%.

	10%	**20%**	**30%**	**40%**	**50%**	**60%**	**70%**	**80%**	**90%**
SR-HTR	0.9157	0.9585	0.9718	0.9799	0.9885	0.9849	0.9877	0.9892	0.9910
Molina [36]	0.9152	0.9580	0.9716	0.9797	0.9884	0.9848	0.9876	0.9891	0.9909
Fan [23]	0.8007	0.9210	0.9144	0.9483	1.0000	0.9601	0.9748	0.9659	0.9762
Singh [2]	0.9855	1.0000	1.0000	0.9963	1.0000	0.9975	1.0000	1.0000	0.9983
Tai [24]	0.9670	0.9855	0.9903	0.9939	1.0000	0.9943	0.9958	0.9963	0.9972
Tong [10]	0.9855	1.0000	1.0000	0.9963	1.0000	0.9975	1.0000	1.0000	0.9983

Figure 12 illustrates the images utilized for the Precision and Recall comparison with the second alteration scheme, which considers the modification of multiple and irregular areas. Moreover, Table 8 shows the average Precision values of SR-HTR and the other state-of-the-art methods for the attacks given in Figure 12.

Table 8. Comparison of *Precision* for multiple and irregular modification detection.

	SR-HTR	**Molina [36]**	**Fan [23]**	**Singh [2]**	**Tai [24]**	**Tong [10]**
Kodak-4	0.9797	0.9782	0.9451	1	0.9939	1
Kodak-10	0.9148	0.9060	0.7573	1	0.9776	1
Kodak-17	0.9784	0.9770	0.7628	1	0.9779	1
Kodak-18	0.9243	0.9177	0.8951	1	0.9884	1

Again, it can be observed that methods [2,10] achieved a higher performance in this test by employing a more significant number of authentication bits. These results are followed by the method [24], then the designed SR-HTR, [36] and finally [23]. Nonetheless, the proposed SR-HTR method accomplished *Precision* values higher than 0.9. It is further demonstrated that the novel SR-HTR method maintains a better balance between authentication and reconstruction capabilities.

Regarding the *Recall* measure, all the methods presented a value of 1.0 for both alteration schemes, including the novel SR-HTR method using the hierarchical authentication method.

To perform a visual comparison, the image Kodak-15 was employed. Its modifications can be observed in Figure 13. This comparison is illustrated in Tables 9 and 10, where Table 9 presents the results for the alteration rates between 20% and 50%, and Table 10 shows the results for alteration rates between 60% and 90%.

Figure 12. Multiple and irregular alterations, (**a**) Kodak-4 original, (**b**) Kodak-10 original, (**c**) Kodak-17 original, (**d**) Kodak-18 original, (**e**) Kodak-4 modified, (**f**) Kodak-10 modified, (**g**) Kodak-17 modified, (**h**) Kodak-18 modified, (**i**) Kodak-4 ground truth, (**j**) Kodak-10 ground truth, (**k**) Kodak-17 ground truth, (**l**) Kodak-18 ground truth.

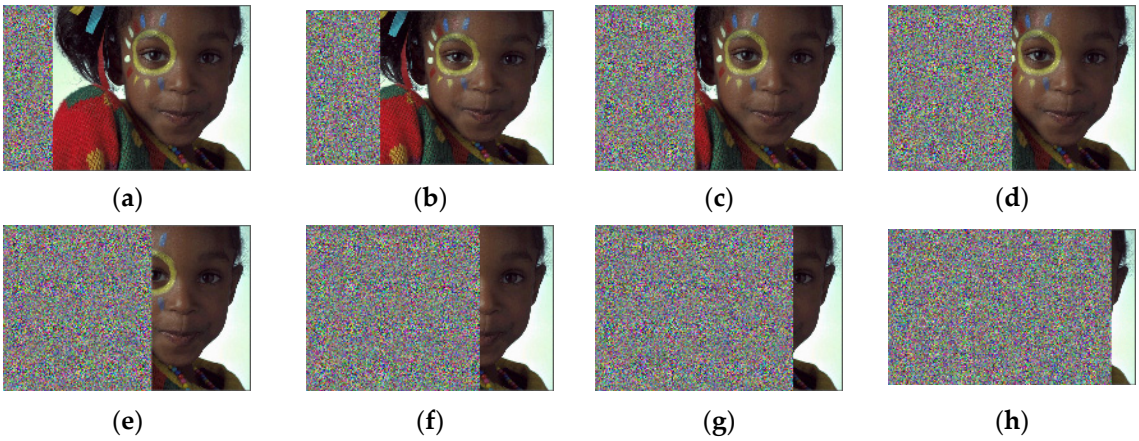
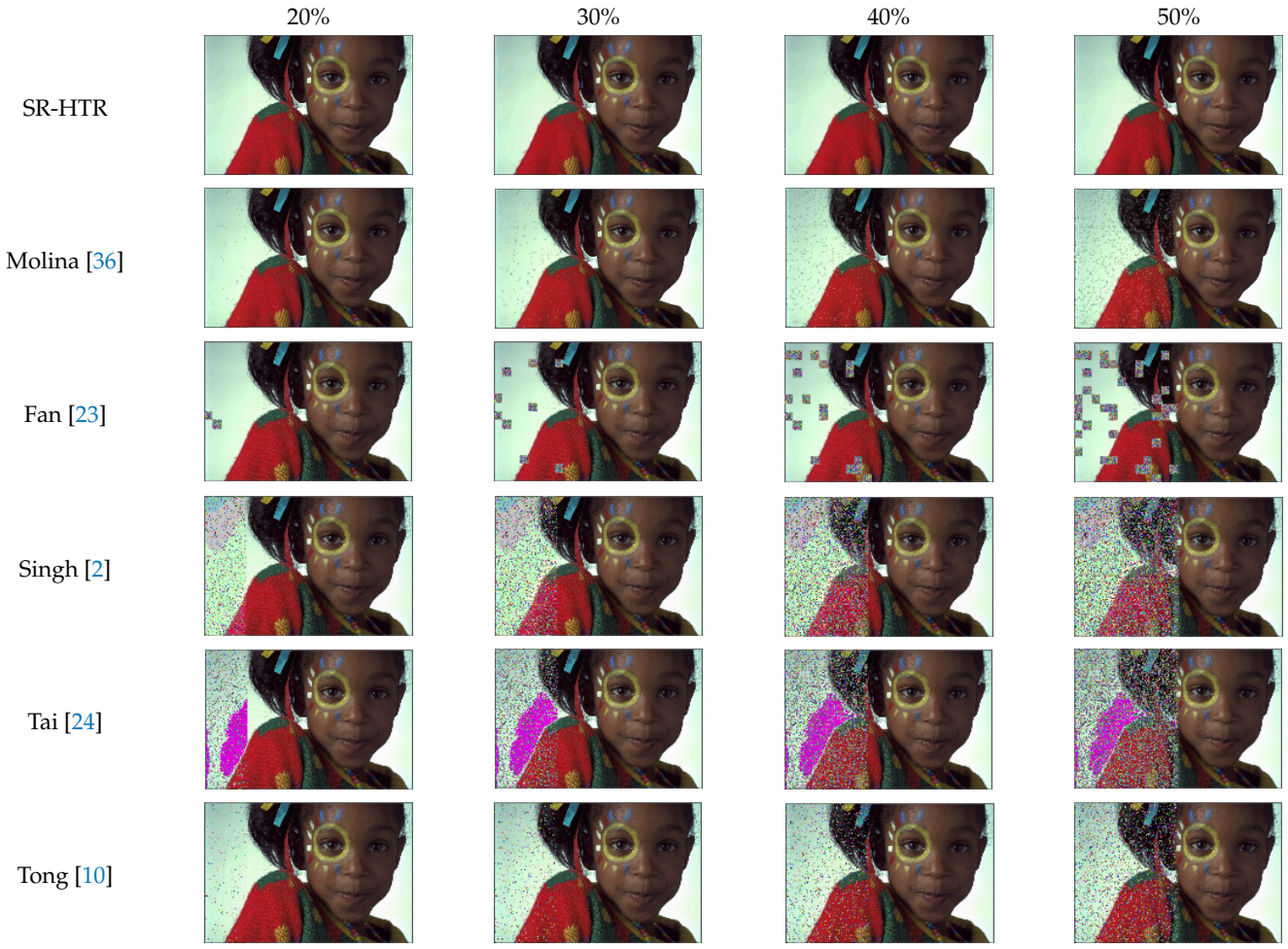


Figure 13. Altered images for Kodak-15, (**a**) 20%, (**b**) 30%, (**c**) 40%, (**d**) 50%, (**e**) 60%, (**f**) 70%, (**g**) 80%, (**h**) 90%.

Table 9. Visual quality comparison for the reconstruction of Kodak-15 using an alteration rate between 20% and 50%.



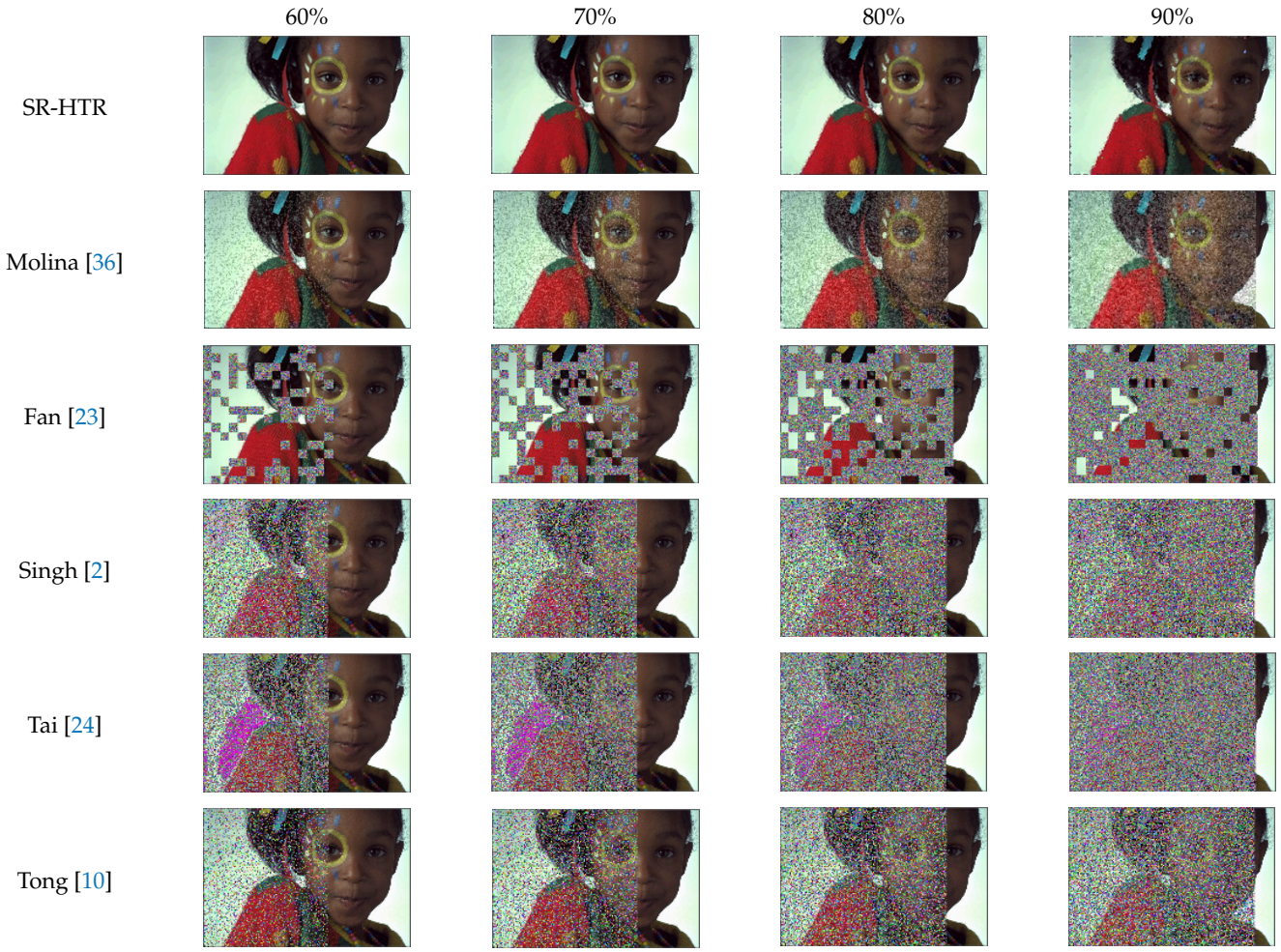
It can be noticed in Table 9 that the designed SR-HTR method presents a better visual quality for alteration rates between 20% and 50%, followed by the procedure [36] that inserts three versions of the reconstruction watermark, and then by [10] that inserts only two. The methods [2,23,24] reconstruct the original content but with a significant loss of quality in the reconstructed image due to the insertion of a single instance of the recovery watermark.

The results given in Table 10 show a dramatic fall in quality for the state-of-the-art methods, while SR-HTR maintains an acceptable visual quality proficiency until 90% when the Kodak-15 image is modified. The second-best results belong to [36], which degrades the image quality but still visibly preserves the original content. This decrease in quality for the state-of-the-art methods is due to the low level of redundancy inserted for the reconstruction watermark, which corresponds to the insertion of up to two copies, such as in [10,23]. However, the framework [23] presents a larger number of contiguous pixels without reconstruction compared with [10] because it uses the SPITH compression algorithm with 32×32 pixel blocks to generate the reconstruction bits. This method is sensitive to noise applied to the blocks. Therefore, the decompression algorithm will result in wrong values for the whole block with minimal modification of the reconstruction bits.

The values of PSNR, SSIM and PSNR-HVS-M for the visual results shown in Tables 9 and 10 are reported in Table 11. It can be observed that for alteration rates greater than 30%, the proposed SR-HTR method presents a better reconstruction performance in the metrics PSNR and PSNR-HSV-M. Furthermore, the process [36] provides a higher reconstruction performance for alteration rates lower than 30%. Regarding SSIM, the best results for 20% and 30% rates belong to [23] due to the SPITH transform utilized in the generation of the

reconstruction bits, which allows a higher quality in reconstructing the blocks. Nonetheless, the SR-HTR method maintains an acceptable performance when more than 40% of the image is modified. It is important to emphasize that the reconstructed image through SR-HTR is not significantly affected by the growth in the alteration rate.

Table 10. Visual quality comparison for the reconstruction of Kodak-15 using an alteration rate between 60% and 90%.



The results displayed in Tables 12–14 represent the average values of PSNR, SSIM and PSNR-HVS-M, correspondingly, for reconstructing the test images using the first alteration scheme. These results reflect the evaluation shown in Tables 9–11, where the proposed method demonstrates more balanced results between the change detection and the reconstruction process. Although a lower performance in change detection was obtained (as shown in Tables 7 and 8), the reconstructed images' quality drastically increased since the proposed method generates an authentication bit for each set of 15 reconstruction bits. The SR-HTR method aims to obtain a better performance in reconstructing large areas of modified pixels without considerably affecting the change detection process and the quality of the watermarked image, as one can see in Table 6. Subsequently, the method [36] achieves the second-best performance due to its insertion process that uses three copies of the watermarks. Methods [10,23] are placed in the third and fourth positions because of their intolerance to high alteration rates and because of the insertion of only two copies of the reconstruction bits. Finally, methods [2,24], which only insert a single version of the reconstruction bits, have demonstrated a lower performance due to their inability to solve the tampering coincidence problem.

Table 11. Comparison of objective quality metrics for the reconstruction of Kodak-15 using alteration rates between 20% and 90%.

		20%	30%	40%	50%	60%	70%	80%	90%
PSNR	SR-HTR	33.295	31.524	29.897	28.836	28.194	27.398	26.459	24.106
	Molina [36]	34.795	31.374	28.282	25.499	22.924	20.705	18.548	16.474
	Fan [23]	29.166	22.849	20.16	17.142	13.604	11.855	9.565	8.295
	Singh [2]	20.603	17.251	14.903	13.059	11.632	10.514	9.452	8.356
	Tai [24]	16.567	14.556	13.312	12.017	10.914	9.971	9.018	7.998
	Tong [10]	27.882	22.728	19.219	16.439	14.196	12.394	10.734	9.101
SSIM	SR-HTR	0.9712	0.9350	0.8920	0.8568	0.8233	0.7875	0.7597	0.7091
	Molina [36]	0.9591	0.9103	0.8397	0.7537	0.6397	0.5177	0.4012	0.3187
	Fan [23]	0.9745	0.9284	0.8858	0.8225	0.6740	0.5515	0.3286	0.1694
	Singh [2]	0.8282	0.7287	0.6271	0.5254	0.4190	0.3148	0.2106	0.1044
	Tai [24]	0.8333	0.7341	0.6333	0.5277	0.4156	0.3092	0.2036	0.0997
	Tong [10]	0.9168	0.8154	0.6964	0.5772	0.4536	0.3366	0.2223	0.1104
PSNR-HVS-M	SR-HTR	30.789	28.894	27.145	26.108	25.474	24.491	23.139	20.164
	Molina [36]	32.717	28.032	24.435	21.457	18.723	16.423	14.236	12.131
	Fan [23]	28.126	21.793	19.231	16.051	12.381	10.613	8.481	7.185
	Singh [2]	18.323	14.821	12.481	10.512	8.963	7.825	6.742	5.535
	Tai [24]	18.879	15.605	13.305	11.32	9.969	9.033	8.042	6.826
	Tong [10]	25.895	20.794	17.082	14.177	11.798	9.895	8.157	6.375

Table 12. Comparison of average PSNR for the reconstruction of the test images with alteration rates between 10% and 90%.

	10%	20%	30%	40%	50%	60%	70%	80%	90%
SR-HTR	35.80	32.72	30.63	29.25	28.14	27.06	26.01	24.82	23.08
Molina [36]	37.28	33.97	31.20	28.71	26.38	23.99	21.77	19.79	18.01
Fan [23]	31.62	29.32	23.75	20.96	18.41	14.92	12.98	10.99	9.63
Singh [2]	26.74	21.76	18.61	16.30	14.54	13.04	11.79	10.70	9.71
Tai [24]	26.34	21.05	17.94	15.71	14.02	12.57	11.36	10.33	9.40
Tong [10]	35.22	28.68	23.98	20.59	17.92	15.61	13.67	11.98	10.46

Table 13. Comparison of average SSIM for the reconstruction of the test images with alteration rates between 10% and 90%.

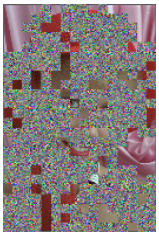
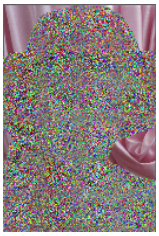
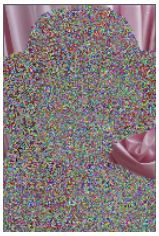
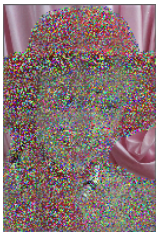
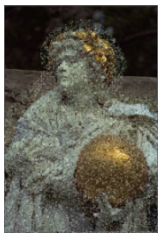
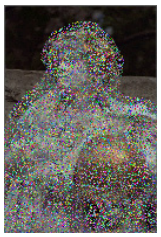
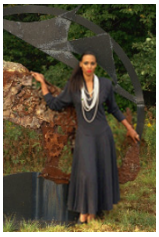
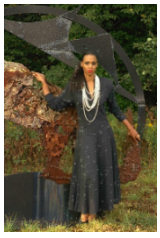
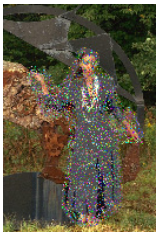
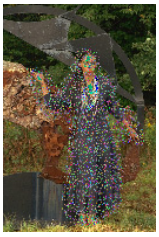
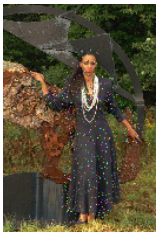
	10%	20%	30%	40%	50%	60%	70%	80%	90%
SR-HTR	0.967	0.935	0.901	0.868	0.833	0.793	0.750	0.696	0.616
Molina [36]	0.972	0.943	0.906	0.852	0.780	0.679	0.561	0.440	0.336
Fan [23]	0.977	0.959	0.919	0.875	0.817	0.676	0.540	0.347	0.167
Singh [2]	0.934	0.841	0.743	0.639	0.537	0.431	0.326	0.220	0.111
Tai [24]	0.942	0.854	0.753	0.648	0.542	0.429	0.319	0.210	0.103
Tong [10]	0.974	0.921	0.834	0.722	0.603	0.478	0.359	0.240	0.122

Table 14. Comparison of average PSNR-HSV-M for the reconstruction of the test images with alteration rates between 10% and 90%.

	10%	20%	30%	40%	50%	60%	70%	80%	90%
SR-HTR	33.29	30.20	28.15	26.70	25.57	24.38	23.03	21.32	18.88
Molina [36]	36.12	32.47	29.10	25.89	23.07	20.26	17.75	15.54	13.58
Fan [23]	31.37	29.33	23.21	20.33	17.73	14.09	12.13	10.15	8.78
Singh [2]	23.79	18.85	15.71	13.36	11.55	9.98	8.67	7.54	6.52
Tai [24]	25.29	19.98	16.77	14.41	12.67	11.27	10.16	9.25	8.43
Tong [10]	34.70	26.87	21.80	18.13	15.25	12.79	10.73	8.94	7.33

The last evaluation is the comparison of the reconstruction proficiency of multiple and irregular alterations. This test was performed using the modifications illustrated in Figure 12. Their corresponded reconstructed images are shown in Table 15. The images' alteration rates are Kodak-4 at 85.65%, Kodak-10 at 26.42%, Kodak-17 at 74.36% and Kodak-18 at 30.97%. The PSNR, SSIM and PSNR-HVS-M values of the reconstructed image compared with the original ones are reported in Table 16.

Table 15. Visual comparison of the reconstructed images for multiple and irregular alterations.

	SR-HTR	Molina [36]	Fan [23]	Singh [2]	Tai [24]	Tong [10]
Kodak-4						
Kodak-10						
Kodak-17						
Kodak-18						

On one hand, it can be observed that the proposed SR-HTR method presents the best objective quality for each reconstructed image. On the other hand, the method given by [23] resulted in better SSIM values for Kodak-10 because the alteration rate is low. However, the visual quality of the reconstructed Kodak-10 given in Table 15 for the proposed method represents an admissible reconstruction.

As can be noticed in Tables 9–16, the novel SR-HTR method demonstrated an excellent performance compared with the other state-of-the-art schemes in terms of regular attacks (as seen in Tables 9–14) and multiple and irregular attacks (shown in Tables 15 and 16).

Table 16. PSNR, SSIM and PSNR-HVS-M for reconstructed images presented in Table 15.

		SR-HTR	Molina [36]	Fan [23]	Singh [2]	Tai [24]	Tong [10]
PSNR	Kodak-4	25.90	19.42	10.01	10.25	9.97	11.22
	Kodak-10	30.21	32.05	24.50	20.27	18.97	26.17
	Kodak-17	26.76	20.95	11.33	11.46	11.20	13.07
	Kodak-18	27.91	28.97	19.17	17.49	17.20	22.85
SSIM	Kodak-4	0.6883	0.3433	0.1961	0.1429	0.1341	0.1548
	Kodak-10	0.9130	0.9197	0.9150	0.7680	0.7818	0.8658
	Kodak-17	0.7661	0.4994	0.3858	0.2650	0.2602	0.2933
	Kodak-18	0.8848	0.8868	0.8553	0.7187	0.7262	0.8174
PSNR-HVS-M	Kodak-4	21.69	14.78	11.21	8.21	11.27	9.23
	Kodak-10	26.52	30.26	24.06	16.67	15.76	23.65
	Kodak-17	23.91	16.77	10.42	8.37	10.27	10.13
	Kodak-18	25.66	27.51	17.68	14.44	14.27	20.61

7. Conclusions

This paper proposes a novel fragile scheme named SR-HTR based on watermarking for color image authentication and self-recovery with high tampering rates. The image protection and extraction method can be implemented in three different variants (1-LSB, 2-LSB or 3-LSB), where it is possible to embed multiple copies (three, six or nine, respectively) of the recovery watermarks, and thus to increase the robustness of the scheme to the tampering coincidence problem. The evaluation of the three embedding–extraction variants of the novel method was carried out. The embedding watermark scheme uses a pseudo-random sequence to embed the recovery watermarks in different blocks to increase the watermarked image's objective quality.

During the evaluation of the results, various alterations were investigated at different tampering rates (from 10% to 90%) with irregular and multiple alterations. Finally, during the recovery process, the 2-LSB embedding scheme was selected as a balanced solution between the quality of the watermarked image and the recovered images' quality for different alterations.

The experimental results have shown good quality for the watermarked images obtained by the novel SR-HTR framework compared with state-of-the-art methods. Additionally, the novel scheme has demonstrated a good performance during the detection of regular, irregular and multiple alterations, resulting in *Precision* and *Recall* metrics higher than 0.9.

The designed SR-HTR has shown excellent performance in reconstructing the alterations at different tampering rates (from 10% to 90%), which is superior to other state-of-the-art methods. Additionally, in cases of multiple and irregular alterations, the novel color image authentication and self-recovery framework has shown an excellent performance, maintaining high objective criteria values as well as great visual perception via the HVS.

As future work, further investigations should be performed to resist different intentional attacks like cropping, scaling and rotation, and unintentional attacks such as JPEG compression. Additionally, for fast processing in real-time environments, we will consider the possibility of designing parallel fragile watermarking schemes implemented in *graphics processing units* (GPU) or multicore *central processing units* (CPU), based on adversarial examples, which have demonstrated remarkable performance in solving similar problems in image and audio domains [42], avoiding weaknesses presented by *deep neural networks* (DNNs) where if an image is transformed slightly, will be incorrectly classified by a DNN even when the changes are small and unnoticed by the human eye [43].

Author Contributions: Conceptualization, R.R.-R., J.M.-G. and B.P.G.-S.; Formal analysis, C.C.-R.; Investigation, R.R.-R.; Methodology, R.R.-R., V.P. and J.M.-G.; Software, C.C.-R. and B.P.G.-S.; Visualization, C.C.-R. and J.M.-G.; Writing—Original draft, B.P.G.-S. and J.M.-G.; Writing—Review and editing, V.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Acknowledgments: The authors would like to thank the Instituto Politecnico Nacional (Mexico), Comision de Operacion y Fomento de Actividades Academicas (COFAA) of IPN and the Consejo Nacional de Ciencia y Tecnologia (Mexico) for their support in this work.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Li, C.; Wang, Y.; Ma, B.; Zhang, Z. A novel self-recovery fragile watermarking scheme based on dual-redundant-ring structure. *Comput. Electr. Eng.* **2011**, *37*, 927–940. [CrossRef]
2. Singh, D.; Singh, S.K. Effective self-embedding watermarking scheme for image tampered detection and localization with recovery capability. *J. Vis. Commun. Image Represent.* **2016**, *38*, 775–789. [CrossRef]
3. Wu, W.-C.; Lin, Z.-W. SVD-based self-embedding image authentication scheme using quick response code features. *J. Vis. Commun. Image Represent.* **2016**, *38*, 18–28. [CrossRef]
4. Qi, X.; Xin, X. A quantization-based semi-fragile watermarking scheme for image content authentication. *J. Vis. Commun. Image Represent.* **2011**, *22*, 187–200. [CrossRef]
5. Chang, C.-C.; Chen, K.-N.; Lee, C.-F.; Liu, L.-J. A secure fragile watermarking scheme based on chaos-and-hamming code. *J. Syst. Softw.* **2011**, *84*, 1462–1470. [CrossRef]
6. Liu, S.-H.; Yao, H.-X.; Gao, W.; Liu, Y.-L. An image fragile watermark scheme based on chaotic image pattern and pixel-pairs. *Appl. Math. Comput.* **2007**, *185*, 869–882. [CrossRef]
7. Chaluvadi, S.B.; Prasad, M.V.N.K. Efficient image tamper detection and recovery technique using dual watermark. In *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*; Institute of Electrical and Electronics Engineers (IEEE): New York, NY, USA, 2009; pp. 993–998.
8. Zhang, X.; Qian, Z.; Ren, Y.; Feng, G. Watermarking with Flexible Self-Recovery Quality Based on Compressive Sensing and Compositive Reconstruction. *IEEE Trans. Inf. Forensics Secur.* **2011**, *6*, 1223–1232. [CrossRef]
9. He, H.; Chen, F.; Tai, H.-M.; Kalker, T.; Zhang, J. Performance Analysis of a Block-Neighborhood-Based Self-Recovery Fragile Watermarking Scheme. *IEEE Trans. Inf. Forensics Secur.* **2011**, *7*, 185–196. [CrossRef]
10. Tong, X.; Liu, Y.; Zhang, M.; Chen, Y. A novel chaos-based fragile watermarking for image tampering detection and self-recovery. *Signal Process. Image Commun.* **2013**, *28*, 301–308. [CrossRef]
11. Qian, Z.; Feng, G. Inpainting Assisted Self Recovery With Decreased Embedding Data. *IEEE Signal Process. Lett.* **2010**, *17*, 929–932. [CrossRef]
12. Qin, C.; Chang, C.-C.; Chen, K.-N. Adaptive self-recovery for tampered images based on VQ indexing and inpainting. *Signal Process.* **2013**, *93*, 933–946. [CrossRef]
13. Li, C.; Wang, Y.; Ma, B.; Zhang, Z. Tamper detection and self-recovery of biometric images using salient region-based authentication watermarking scheme. *Comput. Stand. Interfaces* **2012**, *34*, 367–379. [CrossRef]
14. He, H.-J.; Zhang, J.-S.; Tai, H.-M. Self-recovery Fragile Watermarking Using Block-Neighborhood Tampering Characterization. In *Computer Vision*; Springer International Publishing: Berlin/Heidelberg, Germany, 2009; Volume 5806, pp. 132–145.
15. Dadkhah, S.; Manaf, A.A.; Hori, Y.; Hassanien, A.E.; Sadeghi, S. An effective SVD-based image tampering detection and self-recovery using active watermarking. *Signal Process. Image Commun.* **2014**, *29*, 1197–1210. [CrossRef]
16. Zhang, X.; Wang, S.; Qian, Z.; Feng, G. Reference Sharing Mechanism for Watermark Self-Embedding. *IEEE Trans. Image Process.* **2011**, *20*, 485–495. [CrossRef] [PubMed]
17. Lee, T.-Y.; Lin, S.D. Dual watermark for image tamper detection and recovery. *Pattern Recognit.* **2008**, *41*, 3497–3506. [CrossRef]
18. Zhang, X.; Wang, S.; Qian, Z.; Feng, G. Self-embedding watermark with flexible restoration quality. *Multimed. Tools Appl.* **2011**, *54*, 385–395. [CrossRef]
19. Bravo-Solorio, S.; Nandi, A.K. Secure fragile watermarking method for image authentication with improved tampering localisation and self-recovery capabilities. *Signal Process.* **2011**, *91*, 728–739. [CrossRef]
20. Wang, C.-P.; Wang, X.-Y.; Xia, Z.-Q.; Zhang, C.; Chen, X.-J. Geometrically resilient color image zero-watermarking algorithm based on quaternion Exponent moments. *J. Vis. Commun. Image Represent.* **2016**, *41*, 247–259. [CrossRef]
21. Hsu, L.-Y.; Hu, H.-T. Blind image watermarking via exploitation of inter-block prediction and visibility threshold in DCT domain. *J. Vis. Commun. Image Represent.* **2015**, *32*, 130–143. [CrossRef]

22. Munoz-Ramirez, D.O.; Reyes-Reyes, R.; Ponomaryov, V.; Cruz-Ramos, C. Invisible digital color watermarking technique in anaglyph 3D images. In Proceedings of the 2015 12th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), Mexico City, Mexico, 28–30 October 2015; pp. 1–6.
23. Fan, M.; Wang, H. An enhanced fragile watermarking scheme to digital image protection and self-recovery. *Signal Process. Image Commun.* **2018**, *66*, 19–29. [CrossRef]
24. Tai, W.-L.; Liao, Z.-J. Image self-recovery with watermark self-embedding. *Signal Process. Image Commun.* **2018**, *65*, 11–25. [CrossRef]
25. Chamlawi, R.; Khan, A.; Usman, I. Authentication and recovery of images using multiple watermarks. *Comput. Electr. Eng.* **2010**, *36*, 578–584. [CrossRef]
26. Chamlawi, R.; Khan, A. Digital image authentication and recovery: Employing integer transform based information embedding and extraction. *Inf. Sci.* **2010**, *180*, 4909–4928. [CrossRef]
27. Chamlawi, R.; Khan, A.; Idris, A. Wavelet Based Image Authentication and Recovery. *J. Comput. Sci. Technol.* **2007**, *22*, 795–804. [CrossRef]
28. Qi, X.; Xin, X. A singular-value-based semi-fragile watermarking scheme for image content authentication with tamper localization. *J. Vis. Commun. Image Represent.* **2015**, *30*, 312–327. [CrossRef]
29. Preda, R.O. Semi-fragile watermarking for image authentication with sensitive tamper localization in the wavelet domain. *Measurement* **2013**, *46*, 367–373. [CrossRef]
30. Molina-Garcia, J.; Reyes-Reyes, R.; Ponomaryov, V.; Cruz-Ramos, C. Watermarking algorithm for authentication and self-recovery of tampered images using DWT. In Proceedings of the 2016 9th International Kharkiv Symposium on Physics and Engineering of Microwaves, Millimeter and Submillimeter Waves (MSMW), Kharkiv, Ukraine, 20–24 June 2016; pp. 1–4. [CrossRef]
31. Horng, S.-J.; Rosiyadi, D.; Li, T.; Takao, T.; Guo, M.; Khan, M.K. A blind image copyright protection scheme for e-government. *J. Vis. Commun. Image Represent.* **2013**, *24*, 1099–1105. [CrossRef]
32. Wang, X.-Y.; Liu, Y.-N.; Han, M.-M.; Yang, H.-Y. Local quaternion PHT based robust color image watermarking algorithm. *J. Vis. Commun. Image Represent.* **2016**, *38*, 678–694. [CrossRef]
33. Dutta, T.; Gupta, H.P. A robust watermarking framework for High Efficiency Video Coding (HEVC)–Encoded video with blind extraction process. *J. Vis. Commun. Image Represent.* **2016**, *38*, 29–44. [CrossRef]
34. Wang, R.-Z.; Lin, C.-F.; Lin, J.-C. Hiding data in images by optimal moderately-significant-bit replacement. *Electron. Lett.* **2000**, *36*, 2069. [CrossRef]
35. Ponomarenko, N.; Silvestri, F.; Egiazarian, K.; Carli, M.; Astola, J.; Lukin, V. On between coefficient contrast masking of DCT basis func-tions, CD-ROM. In Proceedings of the Third International Workshop on Video Processing and Quality Metrics for Consumer Electronics, Scottsdale, AZ, USA, 25–26 January 2007.
36. Molina-Garcia, J.; Garcia-Salgado, B.P.; Ponomaryov, V.; Reyes-Reyes, R.; Sadovnychiy, S.; Cruz-Ramos, C. An effective fragile watermarking scheme for color image tampering detection and self-recovery. *Signal Process. Image Commun.* **2020**, *81*, 115725. [CrossRef]
37. Lin, C.-C.; He, S.-L.; Chang, C.-C. Pixel P Air-Wise Fragile Image Watermarking Based on HC-Based Absolute Moment Block Truncation Coding. *Electron.* **2021**, *10*, 690. [CrossRef]
38. Kim, C.; Yang, C.-N. Self-Embedding Fragile Watermarking Scheme to Detect Image Tampering Using AMBTC and OPAP Approaches. *Appl. Sci.* **2021**, *11*, 1146. [CrossRef]
39. Lee, C.-F.; Shen, J.-J.; Chen, Z.-R.; Agrawal, S. Self-Embedding Authentication Watermarking with Effective Tampered Location Detection and High-Quality Image Recovery. *Sensors* **2019**, *19*, 2267. [CrossRef]
40. Kodak Photo CD, Photo Sampler. Available online: http://www.math.purdue.edu/~{}lucier/PHOTO_CD/ (accessed on 1 March 2021).
41. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [CrossRef]
42. Kwon, H.; Yoon, H.; Park, K.-W. Acoustic-decoy: Detection of adversarial examples through audio modification on speech recognition system. *Neurocomputing* **2020**, *417*, 357–370. [CrossRef]
43. Kwon, H.; Kim, Y.; Yoon, H.; Choi, D. Random Untargeted Adversarial Example on Deep Neural Network. *Symmetry* **2018**, *10*, 738. [CrossRef]