International Journal of Multimedia and Ubiquitous Engineering
Vol.10, No.5 (2015), pp.65-76
http://dx.doi.org/10.14257/ijmue.2015.10.5.08

‘Hop Horse’ Image Authentication Scheme

Mona A. M. Fouad and Ahmed Mokhtar A. Mansour

Computers & Systems Department
National Telecommunication Institute
Cairo, Egypt
CTO
Nile Innovations
Cairo, Egypt
mfouad@nti.sci.eg[1]*, Ahmedmokhtar_adu@yahoo.com*[2]

Abstract

This paper proposes a novel scheme for securing and authenticating digital data. Although, the experiments were applied to digital images, it is valid for sound, and video without any modifications of the core scheme. The developed scheme could be applied to the encoded data at the sender and the receiver sides, apart from the compression and decompression processes. The watermark information is generated automatically from the input data and hidden into the Least Significant Bits (LSBs) of specified pixels according to innovative criteria. Experimentally, the received data is authenticated if it is 100% match the sent data, exposing both geometric and signal processing attacks. The hidden watermark is completely invisible, achieving Peak Signal to Noise Ratio (PSNR) more than 67 decibels (dB) for grayscale images that vary in size and content. The developed scheme is evaluated by verifying images of the testing set before and after tampering, showing comparable performance. A comprehensive analysis for the experimental results revealed performance and efficiency of the proposed scheme is presented. An extensive review of previous work concerning image authentication approaches is presented, showing the need for robust and powerful authentication scheme.

Keywords: *Authenticating, Least Significant Bits, Security, Watermarking*

1. Introduction

Wide use of digital contents through public networks implies necessary use of authentication schemes. Watermarking images is a very popular approach for image authentication. It depends on hiding secrete information (watermark) into the image at the sender side to be examined at the receiver side to authenticate the received image. Image quality shouldn’t be affected due to hiding the watermark and on the other hand it should be affected if the image is tampered.

By reviewing the literature, watermarking techniques are grouped into two main categories. The first category is very restrictive announcing the received image as not authenticated if any slight changes have been detected; this called ‘fragile watermarking’. The other category is robust to changes introduced by lossy compression; this called ‘semi-fragile watermarking’.

Image authentication schemes based on watermarking are also classified as blind, semi-blind, or non-blind according to the presence or absence of both/ either the host data (input image) and/or the watermark information [1]. If the original host image is required at the receiver to reliably extract the embedded watermark for authentication, the scheme is non-blind. Semi-blind watermarking scheme uses the watermark information to identify the embedded watermark. In case of the blind watermark scheme, it does not need neither the host image nor the watermark information for authentication.

If we thought as a hacker for a while: we need to manipulate the image and deceive the receiver by not changing the watermark information. This trick could be detected if the receiver has the watermark information or the original image. To have a copy of the original image means that it has been transmitted before. Who knows? It might be manipulated too. Otherwise you may know the watermark information and where it is hidden. What about if the hacker discovered this watermark! that may be used for bunch of images.

Therefore, an efficient authentication scheme should be content based and blind. It is content based, if the watermark information is extracted from the input image and hidden without affecting the image quality. It is blind if the authentication process is done with no prior information about the watermark or the original image.

The proposed authentication scheme exposes both geometric and signal processing attacks using invisible watermarking, achieving more than 67 dB PSNR, depending on the extracted watermark and the image content. A watermark is generated automatically and hidden into the Least Significant Bits (LSBs) of specified positions in a novel way. The first symbol is placed into a predetermined position, while position of each subsequent symbol is positioned at an eclectic step relative to the previous symbol. Authentication is performed by watermarking the received image again by the same way. If the images before and after the second watermarking are identical then the image is authenticated, otherwise it is not-authenticated and the received image is not the original one. The authentication process neither requires the original host image nor the watermark information, so that the proposed scheme is categorized as blind. The only information that should be on the receiver side is the methodology by which the watermark information will be extracted. In this work a specified methodology for watermark extraction is used, but actually it could be done following certain pre-defined protocol that uses different methodologies according to dated schedule for securing the authentication process more.

The watermarking and the authentication processes could be applied apart from any other processing that might take place before or after watermarking/authentication process such as the compression and decompression processes, which means that only lossless compression is allowed.

Globally, two stages were conducted to develop the proposed scheme; the first one is applied to the input image at the sender side to secure it, by hiding the generated signature as an invisible watermark. The other stage is applied to the received image to authenticate or deny it.

The scheme exposes both geometric and image processing attacks. The geometric attack implies image cropping, translation, rotation, resizing, warping …etc. The image processing attack implies re-compression, dithering, re-quantization, filtering …etc. [2]. All these types of attacks as well as the chromatic attack and the Most Significant Bits (MSBs) attack, which changes the MSBs without changing the LSBs that hold the watermark information, could be easily detected by the proposed scheme.

The rest of the paper is organized as follows: An extensive review for the data authentication techniques is presented in section 2. The proposed scheme is particularly described in section 3. The experimental results is presented and analyzed in section 4. Conclusion and further work is existed in section 5.

2. Image Authentication Review

Before going further surveying the image authentication techniques, it is important to mention works that were conducted for hiding data. The work in [3] demonstrates several data hiding codes, focusing on the fundamental principles of the mathematical models. A reversible data hiding algorithm is proposed in [4], utilizing the least occurred pixels and slightly modifies their grayscale values for embedding the watermark information, achieving PSNR of 48 dB for 8-bit grayscale images. Multilevel data hiding techniques are introduced in [5] and extensively applied for digital image and video in [6] showing the availability of adapting the amount of hidden information to the actual noise conditions. The work in [7] proposed a Lossless Authentication Watermarking scheme (LAW), at which the original data is reconstructed from the received watermarked image at the authentication phase. The watermark embedding phase partitions the code space used for storage of image data into two disjoint parts, which together comprise the complete code space. The original image data in the first part is reversibly embedded into the data in the other part without altering its data.

The reviewed image watermarking techniques could be grouped into two main categories. The first is the domain based category, which encompasses techniques applied in spatial or frequency domains. The second is the image based category, which encompasses techniques applied to specific images, such as the binary, grayscale, or color images as well as the compressed and uncompressed images.

The watermarking techniques applied in spatial and frequency domains are reviewed in [8], [9] and [10]. The methods proposed in [9] and [10] concentrated on the frequency and wavelet based techniques, evaluating the surveyed techniques through comparison and performance analyses. Several watermarking techniques applied in spatial domain store the watermarking information into the LSBs [11] – [16]. In [11], the third and fourth LSBs are used to store the watermarking information. In [12], authors used LSBs by inversing the binary values of the watermark text and shifting the watermark according to the odd or even number of pixel coordinates of image before embedding the watermark, achieving PSNR of 48 dB. The proposed works in [13-15], embed cryptographic signatures into specified flipping pixels. Changes applied to these pixels are applied under the condition that “changes should not destroy the connectivity between pixels, and no isolated corner pixels are created. A binary matrix and a weight matrix are used as secret keys to protect the hidden information in [16]. A semi-blind image verification is proposed in [17] by embedding a binary map of a watermark pattern into the original color image, generating a verification key that is used for authenticating the received image.

Watermarking techniques concern compressed images are found in [18-23]. The work in [18] proposed a blind scheme to authenticate compressed images by adding signal-dependent noise that is embedded into the image at the time of acquisition. The authentication process is conducted based on two assumptions: 1) attacked areas in the tampered image are large and connected, 2) the hacker alter the content of interest areas only. These assumptions consider global and scattered manipulation as it is due to transmission error or lossy compression. Authors in [19] extract camera information about quantization tables, Huffman codes, thumbnails, and exchangeable image file format (EXIF) from a JPEG image and create a unique signature that is used to authenticate the received photo. Image alteration is detected by extracting the signature from an image and comparing it to a known database of specified authentic camera signatures. Any mismatch with the image’s EXIF metadata is specified as evidence of tampering. The work in [20] proposed a semi-blind image content authentication scheme by generating the watermark from the round parity of image quantization. Two authentication measures were conducted for measuring the overall similarity between extracted and embedded watermarks as well as the overall clustering level of tampered error pixels those were further integrated to confirm the image content and localize the tampered areas.

In [21], a robust digital image watermarking algorithm have been proposed based on the multiple transform method, Discrete Wavelet Transform (DWT) and Discrete Fractional Random Transform (DFRNT). A watermark is generated by applying the block code encoding adopting the two-dimensional (2D) barcode for hiding information. The generated watermark image is embedded into DWT-DFRNT using quantization technique. A semi-blind watermarking technique that embeds a wavelet compressed watermark image within the least significant bit (LSB) of the original image pixels is introduced in [22] achieving a PSNR of 63 dB. A multipurpose watermarking scheme is proposed in [23] by quantizing the host image's wavelet coefficients and embedding two complementary watermarks that is be blindly extracted from the received image, achieving maximum PSNR of 41.2 dB.

Other approaches were proposed that don't depend on watermarking the transmitted image. A hierarchical approach is applied in [24] to extract an image signature for automatic content authentication achieving acceptable trade-off between robustness and tampering sensitivity. Also a distributed source coding for image authentication is applied in [25] providing an encoded quantized image projection as authentication data and tampering localization using the sum-product algorithm. A Linear Canonical Transform (LCT) is applied in [26] to create a secrete key to authenticate color images, encrypting the primary images individually. In [27], the relationships between discrete cosine transform (DCT) coefficients at the same position in separate blocks of an image are utilized to generate an invariant signature for authenticating jpeg images.

By analyzing the reviewed authentication schemes based on watermarking the original image, it is obvious that whatever the watermark is embedded into the pixels themselves or the quantized data, the problem always appears at the verification stage. Non-blind and semi-blind algorithms could be hacked if the transmitted watermark information and/or the original image is hacked. Concerning blind algorithms based on watermark that are generated from the EXIF information are failed to authenticate the received image if the EXIF and/or the original image is replaced. That is why we thought about extracting the watermark information from the original image itself, embedding it into the LSBs of specified pixels and perform verification by applying the watermarking procedure again to the received image.

Comparing the proposed image authentication scheme and those in the reviewed image authentication schemes, we could conclude that the proposed scheme is simple and powerful preserving the embedded watermark secure and authentication efficiency as it will be explained in the following sections. The scheme is fragile and blind, any slight change of the original image is detected with no need for prior information about the original image or the watermark information. The embedded watermark is invisible achieving PSNR of more than 67.7 dB overcoming all reviewed Peak Signal to Noise Ratio.

3. The Proposed Scheme

The proposed scheme is applied to authenticate uncompressed (or lossless compressed) gray-scale images. However, it could be extended to authenticate color and lossy compressed images as well as any digital content such as sound and video. The scheme is fully automatic composed of two stages; the first one is the watermarking stage that secures the input data at the sender side, while the other stage authenticates the received data, as shown in Figure 1. Both watermarking and authentication stages apply the same procedure (Watermark/Authentication).

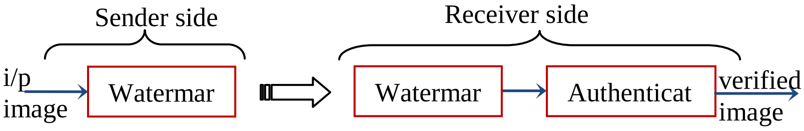


Figure 1. The Authentication Scheme

The watermarking starts by extracting features from the input image and then creates a watermark that is vary in length and content from image to another. A correlation between the image gray levels and the watermark information is generated and then stored into specified locations that are determined in a novel way.

At the sender side, the watermark is generated and embedded into the three LSBs of the specified positions, selected under certain criteria before the image is transmitted. At the receiver side, the authentication process also creates the watermark information from the input image (by the same way it is done at the sender side) and stores it into the three LSBs of the specified positions, selected under the same criteria (as it is done at the sender side).

The image is verified as 'Authenticated' if the difference between the received image and the processed image is zero, because it means that the watermarking is storing the same data at the same positions. In other words, nothing has been changed and the received image is the sent image without any modifications or corruptions. Each of the watermark symbols, the specified locations, and the image contents are correlated.

The watermarking/authenticating procedure is composed of three main steps. The first step extracts features from the input image, and then generates the watermark array. The second step is applied to find the positions that match the watermark symbols within image blocks (in our case it is 8x8 pixels). The third step hides the watermark into the LSBs of specified pixels inside the selected block. The embedding process is repeated until for all symbols of the watermark array.

3.1. Watermarking: Information Generation

As mentioned before, the generation of the watermark might or might not be image based; however, in this paper; the used examples are completely image based. This is done for simplicity, but it is recommended that user defined watermarks might be added to the image based watermark to create hybrid models.

For securing/verifying the examined image in the proposed scheme, a watermark is generated by estimating the image histogram and then used to generate the content of the watermark. Both length and content of the watermark is image-based varies from image to image. The output image is the watermarked image to be transmitted, at the sender side, or to be authenticated at the receiver side.

Experimentally, the image histogram is estimated and pruned to eliminate redundancy. The pruned histogram vector is the seed watermark. This watermark is then hashed to create the real watermark. Our hashing code is shown in Eq. (1).

$$S(i) = rem(H(i),(N-1)), \quad \forall i \tag{1}$$

Where 'S' is the watermark array, 'rem' is the remainder, and 'i' is the ith element of histogram array 'H' of length 'N'.

Technically, to accurately generate the same watermark and store them at the same positions at the sender and receiver sides, histogram estimation should exclude the three LSBs, because they will be changed after watermarking.

3.2. Watermarking: Information Hiding

After generating the watermark information, the positions of the pixel that corresponds to the watermark array elements are realized. They are then hidden into the LSBs of specified locations in the examined image. Two issues are considered while hiding the watermark information. First, the image is divided into blocks of 8x8 pixels and each block cannot carry more than one watermark element. Second, for the 8-bit grayscale images, only three LSBs per pixel is used to identify an address inside the block; in which two pixels are used to identify the position (One is for the x-coordinate and the other is for the y-coordinate).

Before going further storing all watermark array elements, the position of the first element is determined first. It is selected according to Eq. (2).

$$p_1 = rem(kl, b) + 1$$
(2)

Where p1 is the position of the first watermark symbol, *kl* is the length of the watermark array, and b is the number of bits to represent the grayscale levels (b=8, in our case study).

Positions of the subsequent watermark elements, as shown in Figure 2, are determined as follows:

1) Search for the pixel with grayscale level equivalent to the value of the previous element of the watermark array,
2) If it is found, store its x-position and y-position into the three LSBs of the next (or previous, if the block edge is reached) two pixels, else search next block and repeat current step,
3) Get next element, and then repeat the procedure until all elements are stored.

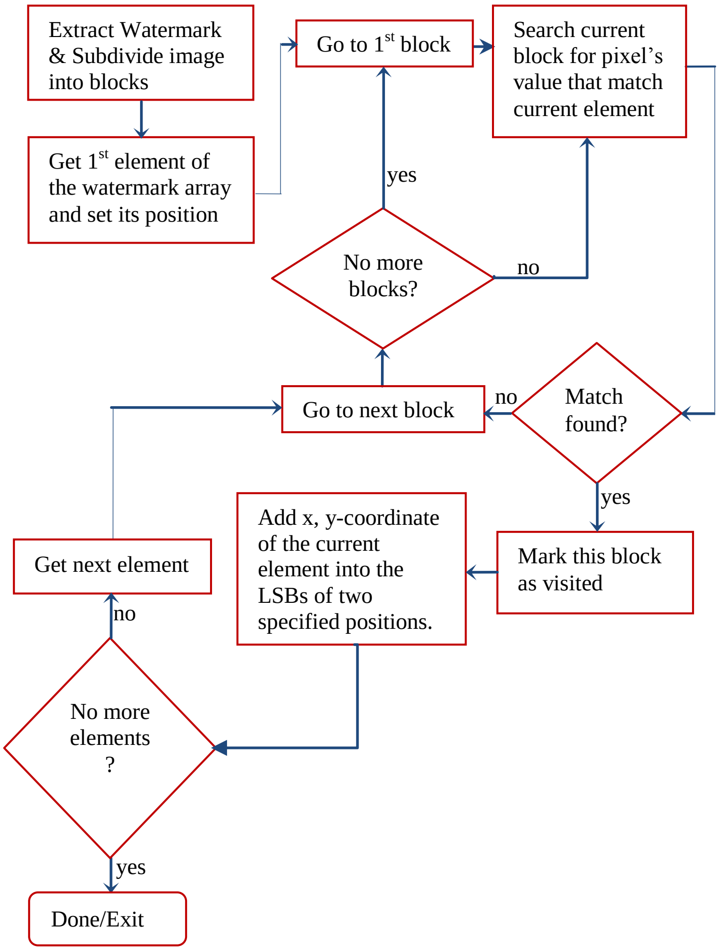

Extract Watermark & Subdivide image into blocks
Go to 1st block
Search current block for pixel's value that match current element
Get 1st element of the watermark array and set its position
yes
No more blocks?
no
Go to next block
no
Match found?
yes
Add x, y-coordinate of the current element into the LSBs of two specified positions.
Mark this block as visited
Get next element
no
No more elements ?
yes
Done/Exit

Figure 2. The Watermarking/Authenticating Algorithm

3.3. Image Authentication

The transmitted image at the sender side is the watermarked image. At the receiver side, the examined image is first watermarked again, by the same way it is done at the sender side. If the watermark is extracted by the same way for the same image and stored at the same positions, then if it is correctly the sent image, then nothing will be changed and the image is ‘authenticated’. If the image is tampered before it is received, then the watermark is not the one that is generated at the sender side, and will be stored at different positions than the original ones, remarking it as ‘not-authenticated’.

4. Experimental Results

One hundred images were examined by the proposed scheme. A subset of the thumbnails of the testing images is shown in Figure 3. The testing set contains all possible varieties that could be in a grayscale image; such as drawings, scanned documents, natural photos, and the full represented images with almost all possible grayscale levels.

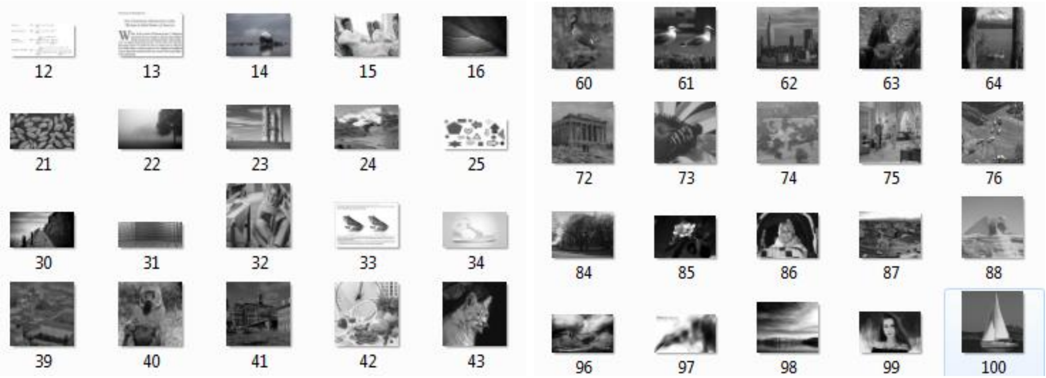


Figure 3. Thumbnails of a Subset of the Testing Set

To examine feasibility and efficiency of the proposed scheme, three main functions were developed to watermark, tamper, and verify. The ‘watermark’ function extracts watermark from the image and then generates the watermarked version of the input image. The ‘tamper’ function attacks the watermarked image and then generates a tampered version of the watermarked image. The ‘verify’ function authenticates the received image and identifies the examined image as authenticated or not. The generated codes were written in MATLAB, on Intel i7-Core Processor of 2.2 GHz and 4GB RAM. Only few seconds are needed to watermark and authenticate an image, depending on its size and number of grayscale levels it has.

As shown in Figure 4, the PSNR achieved is very high, overcoming all in the reviewed literature; this is because the watermark is embedded in the LSBs of specified pixels. You may not be able to distinguish between the original and the watermarked images.

Two kinds of tampers are identified in the ‘tamper’ function. The first manipulates the MSBs. The other, copies certain region(s) and pastes them into another place(s). The experiments show that the proposed scheme is capable of detecting any small modifications, even if it is only in one pixel or one bit. Figure 5 shows a sample image after watermarking, after changing only single pixel, and after changing only single MSB. During authentication stage the watermarked image is verified as “authenticated’ while the tampered images are authenticated as “not-Authenticated”.

Concerning image processing attacks such as filtering and resizing are also examined and the proposed scheme verified them successfully as ‘not-authenticated’.

4.1. Evaluation of the Proposed Authentication Scheme

The first criteria to examine the proposed scheme, is to estimate the PSNR of the watermarked image to examine the visibility of the watermark and whether the two images are similar or not. Two experiments are conducted to examine the proposed scheme. The first experiment applies the authentication procedure, which watermarks the watermarked image, and if the input and output of the authentication procedure are identical, the image is authenticated; otherwise the received image is not the sent one. The second experiment tampers the watermarked image, using our proposed tampering algorithms, and then it is examined using the authentication procedure.

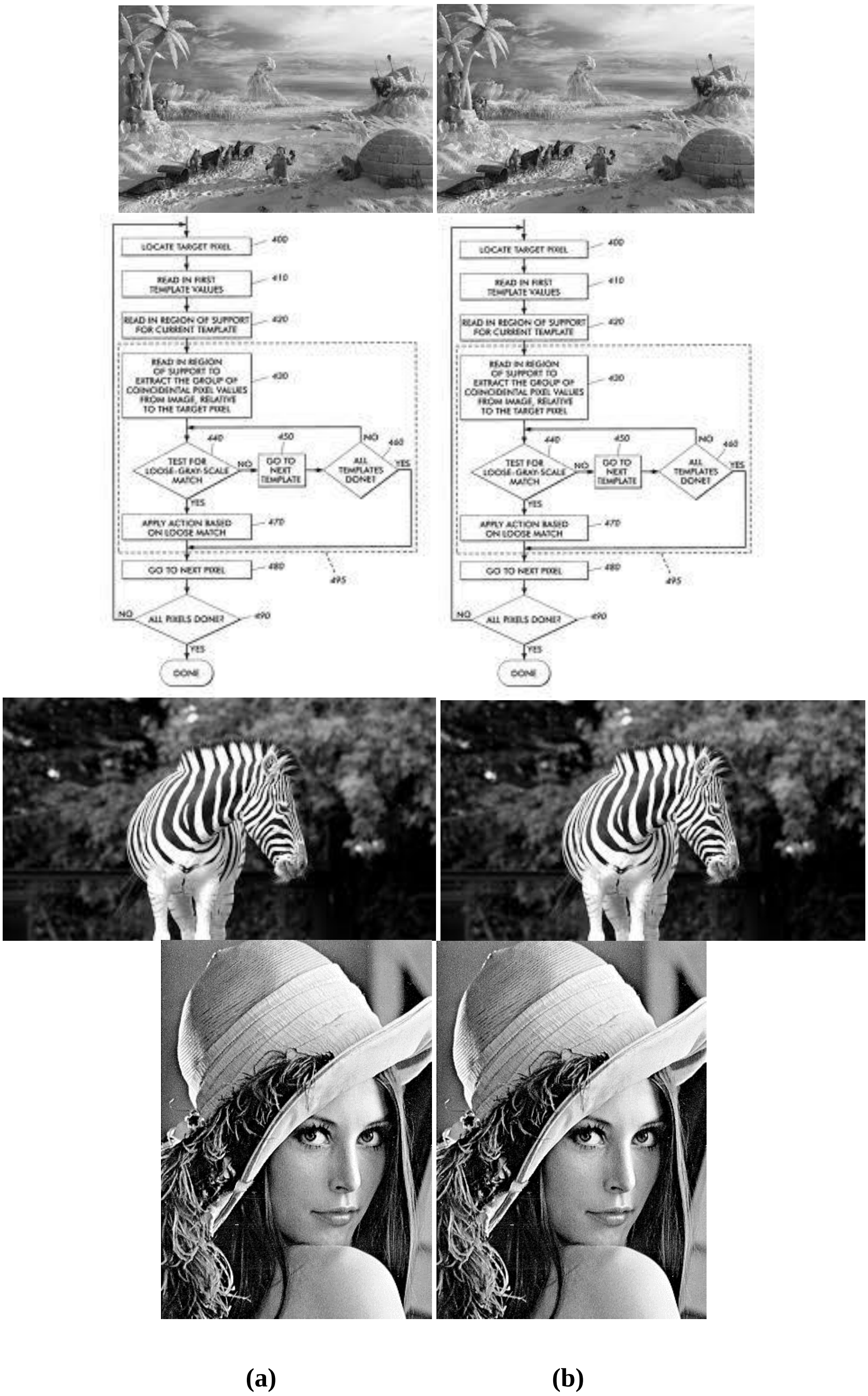
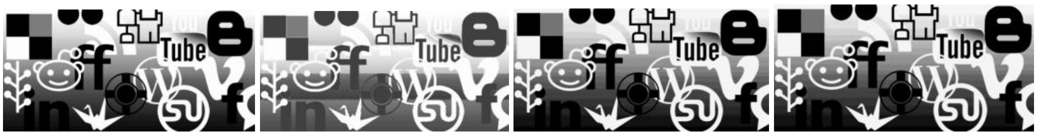

LOCATE TARGET PIXEL
400
READ IN FIRST TEMPLATE VALUES
410
READ IN REGION OF SUPPORT FOR CURRENT TEMPLATE
420
READ IN REGION OF SUPPORT TO EXTRACT THE GROUP OF COINCIDENTAL PIXEL VALUES FROM IMAGE, RELATIVE TO THE TARGET PIXEL
430
440
450
460
TEST FOR LOOSE-GRAY-SCALE MATCH
NO
GO TO NEXT TEMPLATE
ALL TEMPLATES DONE?
YES
NO
YES
APPLY ACTION BASED ON LOOSE MATCH
470
GO TO NEXT PIXEL
480
495
NO
ALL PIXELS DONE?
490
YES
DONE
(a)
(b)

Figure 4. Three Samples from the Testing Set: (a) The Original Image, and (b) The Watermarked Image with Key Lengths 100, 86, 98, 17 and PSNR 69.3, 68.5, 68.5, 80.1 Respectively



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

Figure 5. A Sample Image: (a) the Watermarked, (b) a Tampered Version, Modifying all MSBs, (c) a Tampered Version, Modifying Only One Bit of the MSBs, (d) a Tampered Version, Modifying only Single Pixel in the Whole Image

All images in the testing set is watermarked and successfully verified as ‘Authenticated’, achieving PSNRs in the average of 70 dB, depending on the image representation. Table (1) shows the extracted watermark key-length of subset of original images and the PSNR for each watermarked image. It is seen that the key-length vary from image to image and the PSNR is overcoming all those in the literature. This is because the watermark is embedded into the three LSBs of selected pixels only.

Table 1. A Subset of the Watermark Key Lengths (*kl*) versus the PSNR of a Subset of the Testing Set

kl (symbols)	PSNR (dB)	kl (symbols)	PSNR (dB)
57	70.31	17	80.10
85	68.81	63	68.95
86	68.49	82	68.61
88	68.80	92	72.53
88	69.14	94	76.13
90	71.64	98	68.49
100	69.30	98	67.67
104	68.64	103	69.69
104	67.66	104	69.69
108	69.88	104	76.66

Finally, all images that have been watermarked were tampered using our ‘Tamper’ algorithms and then examined using the ‘verify’ algorithm. This time all images are verified as ‘Not Authenticated’. This is because our authentication scheme verifies not only the LSBs where the watermark is hidden, but it verifies the whole image.

The ‘tamper’ function generates two versions of the tampered images, as shown in Figure 6. The first version is generated by copying a certain region(s) from certain place(s) to another, as in Figure (6-b) (rectangles are surrounding the modified places). The other is generated by modifying the MSBs of all pixels, as in Figure (6-c). All tampered images are successfully verified as ‘Not Authenticated’ when they examined by the ‘verify’ algorithm.

4.2 Evaluation of a set of Benchmark Images

This section shows the result for some standard benchmark images. We used the standard benchmark images downloaded from the internet [28]. This package includes Lena, blond woman, house, lake, etc.

The result of applying the watermarking algorithm in benchmark images are shown in Table 2. As shown in the table, the PSNR for watermarked images is more than 75 in most of the cases. One case showed 74.96 dB as a result of watermarking.

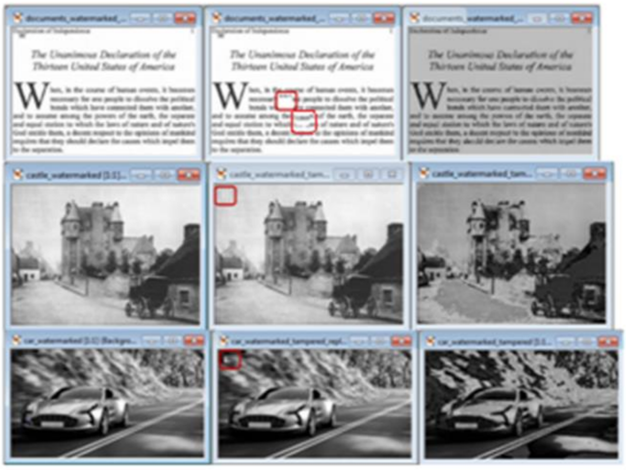


Figure 6. Samples of the Testing Set: (a) The Watermarked Image, (b) The Tampered-replace Image, and (c) The Tampered-MSBs Image

Table 2. Key Lengths (*kl*) Versus the PSNR of the Benchmark Testing Set

image name	cameraman	house	jetplane	lake	livingroom	pirate	woman_darkhair	woman_blonde	walkbridge	peppers
kl	101	103	95	96	98	87	92	85	99	75
PSNR	74.96	75.36	75.50	75.09	75.22	75.96	76.30	76.30	75.80	75.22

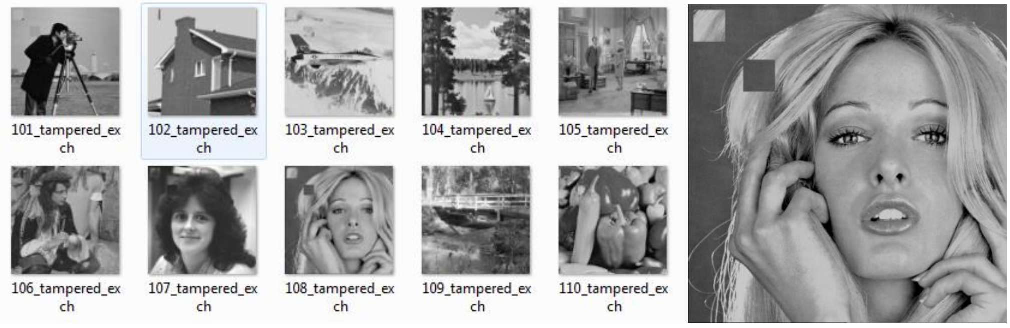


Figure 7. Tampered Images: (a) Thumbnails of the Benchmark Testing Set, (b) The Woman-blond Image

In this experiment, the set of images were tempered without changing the histogram in order to test this type of attack, as shown in Figure 7. This was done by exchanging different regions from the image with each other. After doing so for all images, the authentication algorithm was applied to all images. All images are successfully got the result "not authenticated" although their histograms are identical with their original correspondents.

5. Conclusion

The proposed scheme secures and authenticates grayscale images. A digital watermark is extracted automatically from the input image and embedded into the three LSBs of specified pixels that are selected according to novel criteria. The generated watermark is completely invisible, achieving more than 67.7 dB as PSNR, which overcomes all correspondences in the literature. The received image is authenticated if it is completely matched the sent image.

The efficiency and the performance of the proposed scheme are evaluated by two ways. First, by authenticating the watermarked images that were verified as 'Authenticated' for all tested images. Second, by tampering the watermarked images and then authenticating them; in this case they were verified as 'Not Authenticated'. Global and local modifications as well as the image processing attacks, such as filtering, gray-levels modification, warping, resizing, or cropping, will also be detected by the proposed algorithm easily. Any slight modifications of any bit of any pixel will affect the watermark information that is extracted from the probe image and causes the authentication process to fail.

The scheme is applicable for both color and lossy compressed images. For authentication lossless color images, no changes are needed; the watermark information could be hidden in single color component or distributed among the three components. For lossy compressed images, the watermarking/authenticating processes should be applied apart from the compression/decompression processes.

References

[1] J. J. Eggers and B. Girod, "Blind watermarking applied to image authentication," IEEE Int. Conf. Acoustics, Speech, and Signal Processing, vol. 3, **(2001)**, pp 1977-1980.

[2] S. Voloshynovskiy, S. Pereira, and T. Pun, "Watermark attacks," Erlangen Watermarking Workshop, **(October 1999)**'http://cvml.unige.ch/publications/postscript/99/VoloshynovskiyPereiraPun_eww99.pdf'

[3] Moulin, and Ralf Koetter, "Data-Hiding Codes," Proceedings of IEEE - PIEEE, vol. 93, no. 12, **(2005)**, pp. 2079-2080.

[4] Zhicheng Ni, Yun-Qing Shi, Nirwan Ansari, and Wei Su, "Reversible Data Hiding," IEEE Trans. on Circuits And Systems, vol. 16, no. 3, **(2006)** March, pp. 354-365.

[5] M. Wu and B. Liu, "Data hiding in image and video: Part I—Fundamental issues and solutions," IEEE Trans. on Image Proc., vol. 12, no. 6, **(2003)** June, pp.685-695.

[6] M. Wu, H. Yu, Associate, and B. Liu, "Data Hiding in Image and Video: Part II—Designs and Applications," IEEE Trans. on Image Proc., vol. 12, no. 6, **(2003)** June, pp.696-705.

[7] M. U. Celik, G. Sharma, and A. M. Tekalp, "Lossless Watermarking for Image Authentication: A New Framework and an Implementation", IEEE Transactions on Image Processing, vol. 15, no. 4, April **(2006)**, pp. 1042-1049

[8] V. M. Potdar, S. Han and E. Chang, "A Survey of Digital Image Watermarking Techniques", 3rd IEEE International Conference on Industrial Informatics (INDIN), **(2005)**, pp.709-713.

[9] D. Arya, "A Survey of Frequency and Wavelet Domain Digital Watermarking Techniques. International Journal of Scientific & Engineering Research, vol. 1, no. 2, (**2010)** Novemeber.

[10] J. R. Hernández, M. Amado, and F. Pérez-González, "DCT-Domain Watermarking Techniques for Still Images: Detector Performance Analysis and a New Structure," IEEE Transaction on Image Processing, vol. 9, no. 1, (**2000)** January, pp. 55-68.

[11] A. Bamatraf, R. Ibrahim, M. N. B. M. Salleh, "Digital watermarking algorithm using LSB," Computer Applications and Industrial Electronics (ICCAIE), 2010 International Conference on, (**2010)** December, pp. 155 – 159.

[12] A. Bamatraf, R. Ibrahim and M. Najib Mohd. Salleh, "A New Digital Watermarking Algorithm Using Combination of Least Significant Bit (LSB) and Inverse Bit," Journal of Computing, vol. 3, no. 4, ISSN 2151-9617, (**2011)** April.

[13] H. Yang and A. C. Kot, "Binary Image Authentication With Tampering Localization by Embedding Cryptographic Signature and Block Identifier," IEEE Signal Processing Letters, vol. 13, no. 12, **(2006)**, pp. 741-744.

[14] M. Wu and B. Liu, "Data hiding in binary images for authentication and annotation," IEEE Trans. Multimedia, vol. 6, no. 4, **(2004)**, pp. 528–538.

[15] Y.-C. Tseng, Y.-Y. Chen, and H.-K. Pan, "A secure data hiding scheme for binary images," IEEE Transactions on Communications, vol. 50 , no. 8, **(2002)**, pp. 1227-1231.

[16] H. Yang, A. C. Kot, and J. Liu, "Semi-fragile watermarking for text document images authentication", IEEE International Symposium on Circuits and Systems ISCAS, vol. 4, **(2005)** May, pp. 4002 – 4005.

[17] M. M. Yeung and F. Mintzer, "An Invisible Watermarking Technique For Image Verification", IEEE International Conference on Image Processing Proceedings ICIP'97, vol. 2, **(1997)**, pp. 680 – 683.

[18] S. Ye, Q. Sun, and E.-C. Chang, "Error Resilient Content-based Image Authentication Over Wireless Channel," IEEE International Symposium on Circuits and Systems (ISCAS'05), vol. 3, (**2005)** May, pp. 2707 - 2710.

[19] E. Kee, M. K. Johnson, and H. Farid, "Digital Image Authentication From JPEG Headers," IEEE Trans. on Information Forensics and Security, vol. 6, no. 3, (**2011)** March, pp. 1066 – 1075.

[20] X. Qi and X. Xin, "A quantization-based semi-fragile watermarking scheme for image content authentication", Elsevier Journal of Visual Communication and Image Representation, vol. 22, **(2011)**, pp. 187–200.

[21] M. Kim1, D. Li2 and S. Hong, "A Robust Digital Watermarking Technique for Image Contents based on DWT-DFRNT Multiple Transform Method", International Journal of Multimedia and Ubiquitous Engineering-SERSC, vol. 9, no. 1, **(2014)**, pp. 369-378.

[22] M. F. Al-Hunaity, S. A. Najim and I. M. El-Emary, "Colored Digital Image Watermarking using the Wavelet Technique", American Journal of Applied Sciences, vol. 4, no. 9, **(2007)**, pp. 658-662.

[23] C.-S. Lu, and H.-Y. M. Liao, "Multipurpose Watermarking for Image Authentication and Protection," IEEE Trans. On Image Processing, vol. 10, no. 10, (**2001)** October, pp. 1579-1592.

[24] X. Wang, N. Zheng, J. Xue, and Z. Liu, "A Novel Image Signature Method for Content Authentication," The Computer Journal. vol. 55, no. 6, **(2012)**, pp. 686-701.

[25] Y.-C. Lin, D. Varodayan, and B. Girod, "Image Authentication Using Distributed Source Coding," IEEE Trans. On Image Processing, vol. 21, no. 1, **(2012)**, pp. 273-283.

[26] S. Keshari, M. Alam, and S. G. Modani, "Color image authentication scheme in Linear Canonical Transform Domain," International Conference on Recent Advances in Information Technology (RAIT), **(2012)** March.

[27] C.-Y. Lin and S.-F. Chang, "A Robust Image Authentication Method Distinguishing JPEG Compression from Malicious Manipulation," IEEE Trans. on Circuits and Systems of Video Technology, vol. 11, no. 2, **(2001)** August, pp. 153-168.

[28] "Standard" test images 'http://www.imageprocessingplace.com/root_files_V3/image_databases.html