Computers, Materials & Continua
DOI:10.32604/cmc.2021.018660
Article

Tech Science Press

Local Features-Based Watermarking for Image Security in Social Media

Shady Y. El-mashad[1], Amani M. Yassen[1], Abdulwahab K. Alsammak[1] and Basem M. Elhalawany[2,*]

[1]Department of Computer Systems Engineering, Faculty of Engineering at Shoubra, Benha University, Egypt
[2]Department of Communications and Electronics Engineering, Faculty of Engineering at Shoubra, Benha University, Egypt
[*]Corresponding Author: Basem M. Elhalawany. Email: basem.mamdoh@feng.bu.edu.eg
Received: 16 March 2021; Accepted: 21 April 2021

Abstract: The last decade shows an explosion of using social media, which raises several challenges related to the security of personal files including images. These challenges include modifying, illegal copying, identity fraud, copyright protection and ownership of images. Traditional digital watermarking techniques embed digital information inside another digital information without affecting the visual quality for security purposes. In this paper, we propose a hybrid digital watermarking and image processing approach to improve the image security level. Specifically, variants of the widely used Least-Significant Bit (LSB) watermarking technique are merged with a blob detection algorithm to embed information into the boundary pixels of the largest blob of a digital image. The proposed algorithms are tested using several experiments and techniques, which are followed by uploading the watermarked images into a social media site to evaluate the probability of extracting the embedding watermarks. The results show that the proposed approaches outperform the traditional LSB algorithm in terms of time, evaluation criteria and the percentage of pixels that have changed.

Keywords: Digital watermarking; LSB; social media; blob detection; image security

1 Introduction

Social media has become essential for human interaction all over the globe, where the users create, exchange, and share information on the internet using electronic devices such as computers, wearable devices, and smartphones [1]. Every day the number of social media users are dramatically increase according to the latest statistics [2]. Due to this increase, many digital images have been uploaded into social media. Therefore, a various types of image security issues appeared, which includes modification, illegal copying, identity fraud, copyright protection and ownership of images [3].

Hence, there is a strong need to protect the uploaded images from these security issues. Several techniques based on the digital watermarking have being used and developed to achieve this goal. The digital watermarking is the process of hiding or embedding the digital information inside another digital information without affecting the visual quality of any of them such as: text, image, video, audio and metadata as well that can be extracted [4]. In contrast to encryption, the watermarking allows the user to view, access and get the original digital information [4].

The watermarking algorithm is usually used to achieve the authenticity and integrity by performing two steps, namely the embedding and the extraction. In image watermarking, the original image is known as the cover image or the carrier, while after applying the watermarking process the output image is known as the watermarked image [5]. Fig. 1 shows the block diagram of image watermarking, where the embedding process is done by applying a watermark to the original image using a secret key. On the other hand, the extraction process exploits the same secret key to extract the watermark and retrieve the original image.

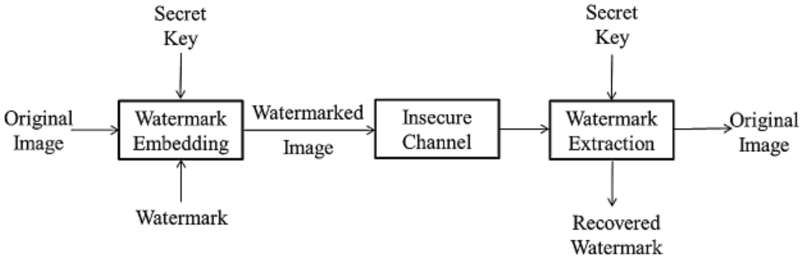

Figure 1: Digital image watermarking

Conventionally, image watermarking techniques work in two domains, namely the spectral domain and the spatial domain. In spectral domain watermarking, the original image is transformed into a group of frequency coefficients [5]. This type is very robust; however, it usually exhibits high complexity. The operations applied to the lower frequencies instead of higher frequencies to avoid the problem of losing frequencies in case of compression or scaling [4]. The most commonly used algorithms in spectral domain are: Discrete Wavelet Transform (DWT), Discrete Cosine Transform (DCT), and Discrete Fourier Transform (DFT).

On the other hand, spatial domain watermarking operations are simple and directly applied to the pixels without transformation as in the former type. The most commonly used algorithm in spatial domain is the Least Significant Bit (LSB). LSB is a simple and fast watermarking approach, which converts the image regardless of its type into gray scale image. Each pixel is represented by 1 byte, where the last bit, i.e., the right most bit, contains the least significant information as shown in Fig. 2.

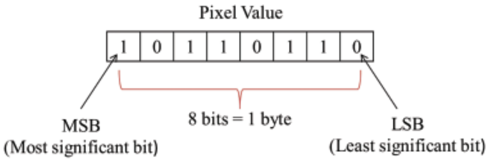

Figure 2: Least significant bit

The LSB watermarking algorithm replaces this bit with one bit of the secret data, which will be hidden inside the image. Since the replaced bit is the least significant one, there will be no obvious visual change in the image [6]. Fig. 3 shows the block diagram of LSB embedding process, where one-pixel value is presented, at which the last bit (i.e., LSB) has a value 1. This value has been replaced with the first bit in the secret message which is 0. Therefore, the new value of the last LSB has become 0 instead of 1.

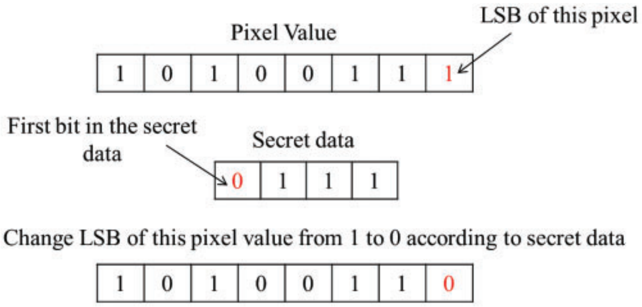
LSB of this pixel
Pixel Value
1 0 1 0 0 1 1 1
First bit in the secret data
Secret data
0 1 1 1
Change LSB of this pixel value from 1 to 0 according to secret data
1 0 1 0 0 1 1 0

Figure 3: Lsb embedding process

- We proposed a local feature-based watermarking technique, which is used in tandem with variants of the traditional LSB algorithm to increase the security level.
- To improve the robustness of the proposed technique, we have implemented a two-bits based LSB algorithm instead of the conventional one-bit based algorithm.
- We investigate the performance of the proposed algorithm in terms of time, evaluation criteria and the percentage of pixels that have changed.

The rest of the paper is organized as follows. In Section 2, we present a literature review on the state-of-the-art watermarking techniques. Then, we introduced the proposed method and experiments design in Sections 3 and 4, respectively. Finally, the paper is concluded in Section 5.

2 Related Works

The digital watermarking provides a protection of digital contents such as: images, videos and music from any illegal use. In the last years, digital watermarking has been combined with different algorithms to increase its security and robustness [7]. Media watermarking has become an active research area. The authors in [8] used LSB algorithm to embed the watermark into the digital image. The third and the fourth LSB are used for hiding the data. Their experimental results show a better quality of the watermarked image. In [9], the authors presented a combination of LSB algorithm and edge detection technique for image stenography. The edge pixels of the image have been used to hide the secret data, which leads to a higher security level.

Also, [3] presented a new watermarking scheme to embed the selected metadata into the digital image. The implementation is done on both visible and invisible watermarking. Within this experiment, the images have been uploaded into four different social media sites. It shows that the compression method of social media sites can change the pixel values of the image. Besides that, [10] proposed a new image stenography method based on extracting the three rightmost LSBs of the image pixels. Then, performing the X-OR operation between the first bit, the third bit and also between the second bit, the third bit. Based on the results of these operations, a +1 or −1 modification applied to the pixel value of the stego image. The proposed method was better when compared to LSB algorithm according to [10].

In addition, [11] presented a new scheme of transform domain JPEG image stenography. The secret data is compressed twice. First, compressed by removing the weak words from the data and then by using the Huffman lossless compression technique. The compressed data is embedded into the original image based on the modulus three of the difference between DCT coefficients of the image. This scheme is proven by [11] to significantly reduce the number of changes in the cover image.

Also, [12] presented a new stenography technique inside RGB color space. First, the cover image was flipped to embed secret message into it. Then, it was divided into the red, green and blue channel. The blue channel was used for embedding secret message which were divided into four sub images and shuffled by using a magic matrix. Before embedding, the secret message was subtracted from the corresponding pixel values of the red channel. Then, the authors in [12] selected 8 bits of secret message to be embedded into the four sub images. This technique achieves enhanced security according to [12].

Eventually, [13] proposed a hybrid combination of watermarking techniques. A local feature based watermarking scheme with a traditional transform domain-based watermarking. Watermarks are embedded twice. First, the KAZE features regions have been selected and watermarks are embedded into significant-bit-planes of these regions by modifying their histogram. Then, the Integer wavelet transform-Singular value decomposition (IWT-SVD) domain has been used to embed watermarks again by modifying the entries of the left singular vector metrics. This method was sufficient 100 robust according to [13].

3 Proposed Method

The main concept in this work is to exploit the advantage of different image processing techniques to embed the secret data in the original image. A new image watermarking technique is introduced based on the LSB algorithm. The proposed technique uses a combination of the LSB algorithm and image processing techniques. This combination enhances the performance of the traditional LSB algorithm. Additionally, it increases the level of image security through online social media sites without affecting the visual quality of the image [4].

Specifically, the secret data will be hidden into a variable selected area or pixels not in the whole image, which increases the robustness of the system. In this technique, the secret data is embedded into the largest blob of the original image. Therefore, a blob detection technique is used to extract the existing blobs in the original image including the largest one. One main objective of this work is to improve the robustness of the proposed algorithm. To cope with this objective, we implemented a modified two-bits based LSB algorithm, at which the last two bits are used for hiding the secret data instead of the last bit only. In the following subsections, we explain the details of the blob detection, embedding algorithm, and the extraction algorithm, respectively.

3.1 Blob Detection

Generally, a blob stands for binary large object, which is composed of a set of connected pixels or points [14]. These pixels are considered to be similar to each other in a binary image. To extract a blob, the connectivity of neighbor pixels is tested to detect similarity [15]. There are many different types of connectivity such as the 4-connectivity and 8-connectivity patterns, which are shown in Fig. 4. In this work, we adopted the 8-connectivity to achieve more accurate results since the 8-connectivity is more precise, however, it imposes higher computations than the 4-connectivity [14]. In image processing, blob detection methods are aimed to disclose regions or points in a digital image that differ in properties compared to the surrounding regions [16–18]. These properties included color or brightness. Fig. 5 shows blob detection examples.

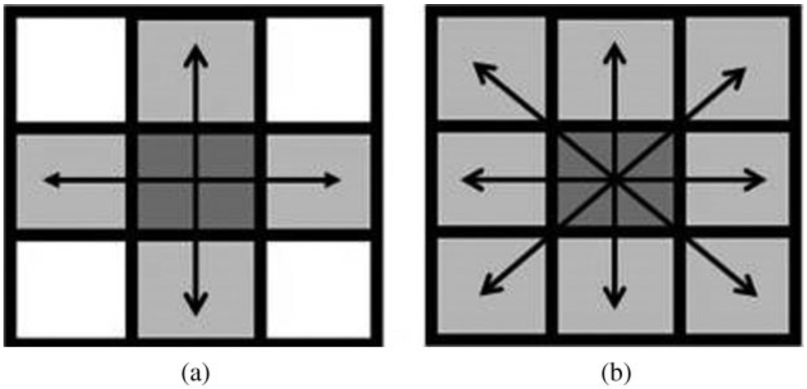

Figure 4: The types of connectivity (a) A 4-connectivity pixel (b) An 8-connectivity pixel

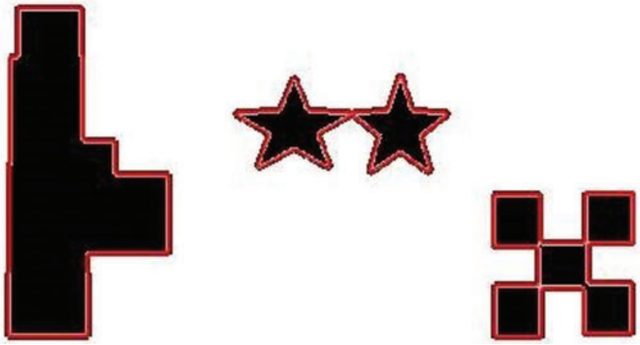

Figure 5: Blob detection examples

In the proposed approach, the secret data is embedded into the boundary pixels of a specific blob only for example the largest blob. This will increase the level of security and robustness. In addition, one than a blob can be chosen for the embedding process. In this technique, the secret watermark is provided into the original image without any visual changing in it.

3.2 Embedding Algorithm

The embedding algorithm has three inputs: a cover image, a secret data and a secret key. It is done as follows:

(1) Convert the secret data into its binary representation.
(2) Convert the cover image into a binary image.
(3) Extract the largest blob of the binary image.
(4) Detect the boundary pixels of the largest blob.
(5) Embed the data bit stream into the LSB of each boundary pixel.
(6) Complete the embedding until the secret data bit stream is finished. The embedding algorithm can be done on more than one bit.

Fig. 6 shows the flowchart of the proposed watermarking framework.

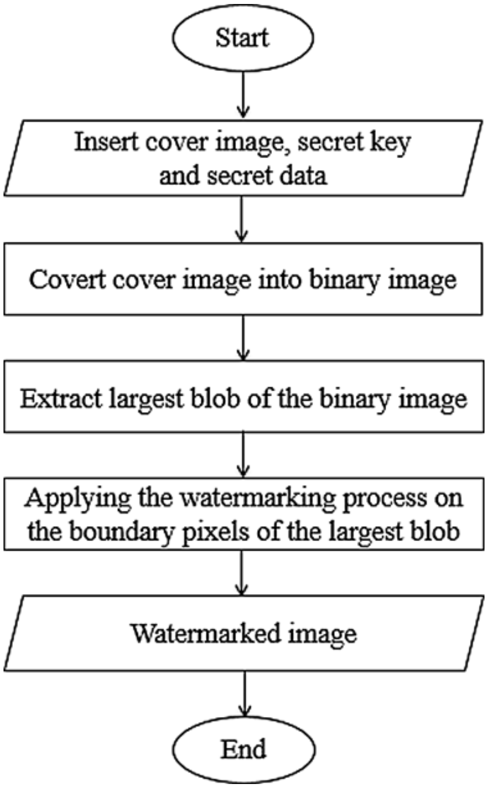
Start
Insert cover image, secret key and secret data
Covert cover image into binary image
Extract largest blob of the binary image
Applying the watermarking process on the boundary pixels of the largest blob
Watermarked image
End

Figure 6: The framework of the proposed method

3.3 Extraction Algorithm

The extraction algorithm has two inputs: a watermarked image and a secret key. It is done as follows:

(1) Convert the watermarked image into a binary image.
(2) Extract the largest blob of the binary image.
(3) Detect the boundary pixels of the largest blob.
(4) Extract each LSB of the boundary pixels.
(5) Concatenate all extracted bits to get a stream of bits.
(6) Convert the extracted bits into its equivalent characters.

4 Experimental Results

In the following sub-sections, we describe the dataset of the secret watermarks and the test images used in our experiments. Additionally, we explain the evaluation criteria used for evaluating the proposed algorithm and the experiment design. Finally, we elaborate in explaining the deployment of the watermarked images on a popular social media site for testing the performance of the proposed algorithms.

4.1 Data Set

The data set used in the experiments consists of two parts. The first part contains four secret watermarks with different lengths, from 46 to 221 characters, as shown in Tab. 1. In this paper, the watermarks are added as data related to social media. The first watermark embeds name, age and gender of the owner of the image. In the second watermark, id, location, and date are added next to the first watermark. The third watermark describes the image. Finally, a caption has been added as a fourth watermark. The second part contains three images including "Tiffany" with size 512 × 512, "Cablecar" with size 512 × 480 and "Barbra" with size 512 × 512 as shown in Fig. 7.

Table 1: Dataset of secret watermarks

No	Watermark message	Message length
W1	Name, age, gender	46 character
W2	Name, age, gender, id, location, date	80 character
W3	A description of the image	119 character
W4	Caption: Whenever you organize a space for its intended purpose....	221 character

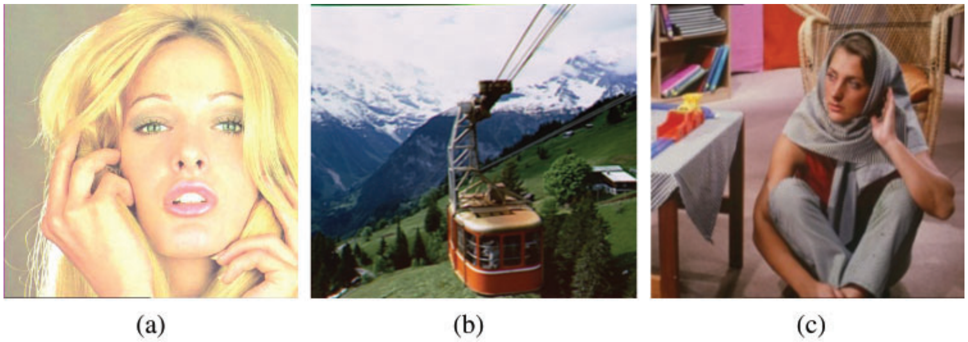

Figure 7: Data set of cover images (a) Tiffany.jpg (b) Cablecar.bmp (c) Brabra.png

Several parameters can be adopted to measure the quality of a watermarked image such as the Peak Signal to Noise Ratio (PSNR), the Mean Square Error (MSE), and the structural similarity index (SSIM). Those parameters can be defined as follows:

4.1.1 PSNR

It used to evaluate the similarity between the original image and the watermarked image. Additionally, it measures the quality of the watermarked images, where the PSNR should be high for a reliable technique. The values of PSNR ranges between [0, +∞] [13]. The general PSNR Eq. (1) is defined as follows:

$$PSNR = 10.log1\left(\frac{MAX_I^2}{MSE}\right) \tag{1}$$

where MAX is the maximum value that a pixel can have (i.e., 255 for a grayscale image).

4.1.2 MSE

It shows the difference between the original image and the watermarked image. However, there is no exact value for MSE. A lower MSE indicates that the watermarked image is more similar to the original image and 0 means the method is perfect [19]. Hence, The MSE should be low for a reliable technique. The MSE is defined in Eq. (2) as:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2 \tag{2}$$

where m and n are the number of rows and columns in the original image. I is the original image and K is the watermarked image.

4.1.3 SSIM

It measures the perceptual difference between the original image and the watermarked image. The values of SSIM ranges between +1 to −1 [12], 1 means that there is a perfect match the watermarked image with the original one. However, values 0.99, 0.98 and 0.97 means a good technique. The SSIM is expressed mathematically as follows in Eq. (3):

$$SSIM(I, I_w) = \frac{(2\mu_I \mu_w + C_1)(2\delta_{Iw} + C_2)}{\left(\mu_I^2 + \mu_w^2 + C_1\right)\left(\delta_I^2 + \delta_w^2 + C_2\right)} \tag{3}$$

where I is the original image and Iw is the watermarked image. μ_I and μ_w are the mean luminance values of I and Iw, respectively. δ_I and δ_w are their standard deviations. δ_{Iw} is the covariance between I and Iw. C_1 and C_2 are positive constants.

4.2 Experimental Design

Based on the evaluation criteria, the experiments have been conducted using MATLAB 9.6.0 on a core i7 CPU up to 4.5 GHZ, with 16 GB memory. Each experiment will embed one of the secret watermarks into the boundary pixels of the largest blob for each cover image. In this paper, many different methods for embedding secret watermark are introduced. In the first experiment, the 1st LSB has been used only for the embedding process, while in the second experiment the 2nd LSB has been used only for the embedding process. In the third experiments, the 1st and 2nd LSBs are used for the embedding process, where half of the watermark has been embedded in the 1st LSB and the other half has been embedded in the 2nd LSB. Finally, the last experiment is based on selecting either of the 1st LSB or the 2nd LSB for the embedding process. If the position of the watermark bit is even, the watermark bit will be embedded in the 1st LSB, and if the position of the watermark bit is odd, then the watermark bit will be embedded in the 2nd LSB. Using these different embedding methods will lead to more reliability. Tab. 2, describes the different methods of embedding.

Table 2: Methods of embedding process

Embedding name	Embedding method
E1	1st LSB
E2	2nd LSB
E3	1st LSB and 2nd LSB
E4	1st LSB or 2nd LSB

The experiments have been applied to different types of images to ensure the correctness of the proposed method with all types of images. Tab. 3 shows the results for these experiments into "Tiffany.jpg", where the boundary pixels of the largest blob are 2044 pixels. The percentage of boundary pixels being used for these experiments are 0.00779, which is a little higher compared to 0.00674 of the conventional LSB. The slight increase is due to the presence of unused pixels in the largest blob depending on the length of the embedded data. As shown in Tab. 3, we compare the MSEs, PSNRs, SSIMs, Insertion time and Retrieval time in tiffany image by using the four embedding methods. The results show that the PSNR values are high for the four embedding methods, the MSE values are low and the SSIM values are significant: 0.9999. Fig. 8, shows the watermarked image and the original image of Tiffany.

Table 3: Experimental results of tiffany.jpg

Secret watermarks	Embedding method	MSE	PSNR	SSIM	Insertion time (s)	Retrieval time (s)
W1	E1	0.00037	82.35985	1	0.529	0.133
	E2	0.00132	76.90004	1	0.557	0.138
	E3	0.00091	78.51409	1	0.593	0.144
	E4	0.00101	78.08374	1	0.571	0.142
W2	E1	0.00067	79.81200	1	0.553	0.139
	E2	0.00247	74.20045	1	0.599	0.143
	E3	0.00150	76.35023	1	0.613	0.147
	E4	0.00171	75.78407	1	0.610	0.150
W3	E1	0.00120	77.31933	1	0.604	0.144
	E2	0.00425	71.83956	0.999	0.632	0.148
	E3	0.00302	73.31799	1	0.674	0.153
	E4	0.00290	73.50235	1	0.669	0.151
W4	E1	0.00195	75.22350	1	0.638	0.149
	E2	0.00703	69.65859	0.999	0.670	0.155
	E3	0.00526	70.91426	0.999	0.701	0.159
	E4	0.00475	71.35753	0.999	0.695	0.158

In addition, Tab. 4 gives the results of these experiments into "cablecar.bmp" image, where the boundary pixels of the largest blob are 3076 pixels. The percentage of boundary pixels being used for these experiments are 0.011734. Based on the results shown, the PSNR values are also high, the MSE values still low and the SSIM values are significant using the different embedding methods. Fig. 9, shows the watermarked and the original image of "cablecar.bmp".

Finally, as shown in Tab. 5, it shows the results of these experiments into Barbra.png. Where the boundary pixels of the largest blob are 6517 pixels and the percentage of boundary pixels being used for these experiments are 0.02486. Unlike the previous two images, the Barbra image had achieved perfect results. The SSIM values indicate a perfect match between the original image and the watermarked image, it was 1 in every experiment. PSNR values in W1-E1 & W2-E2 still high. Also, MSE values still low. Fig. 10 shows the watermarked image and the original image of Barbra.

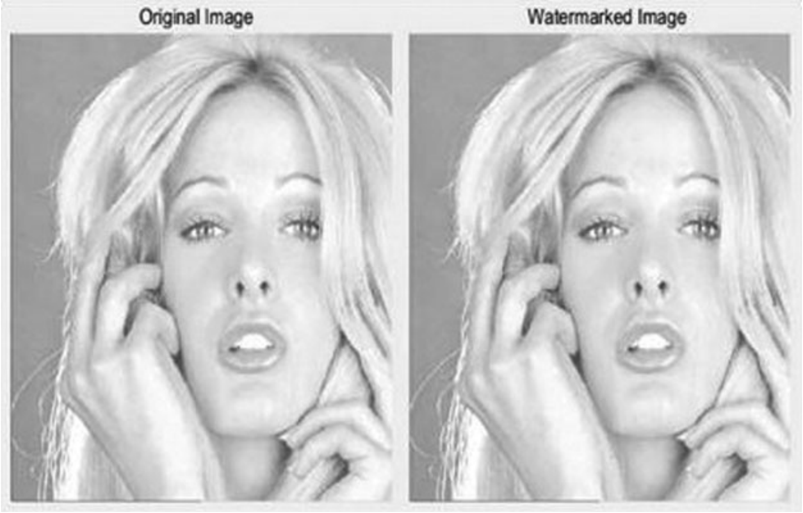

Figure 8: The original image and watermarked image of Tiffany

Table 4: Experimental results of cablecar.bmp

Secret watermarks	Embedding method	MSE	PSNR	SSIM	Insertion time (s)	Retrieval time (s)
W1	E1	0.00041	81.90754	1	0.530	0.132
	E2	0.00138	76.72112	1	0.559	0.140
	E3	0.00103	77.98757	1	0.589	0.1450
	E4	0.00095	78.32523	1	0.570	0.144
W2	E1	0.00696	79.70595	1	0.567	0.138
	E2	0.00244	74.25440	1	0.597	0.143
	E3	0.00183	75.48453	1	0.621	0.147
	E4	0.00164	75.96136	1	0.615	0.149
W3	E1	0.00121	77.29375	1	0.623	0.145
	E2	0.00491	71.21524	1	0.650	0.150
	E3	0.00343	72.77763	1	0.668	0.153
	E4	0.00325	73.01044	1	0.657	0.152
W4	E1	0.00203	75.03753	1	0.643	0.151
	E2	0.00825	68.96523	1	0.675	0.155
	E3	0.00622	70.18900	0.999	0.713	0.159
	E4	0.00528	70.89982	0.999	0.710	0.157

4.3 Security Measurement

The security of the watermarked images is tested by applying cropping attacks. Cropping attacks are usually harmful; therefore, it is used to test the robustness of the watermarking algorithms. Tab. 6 shows the results of these experiments.

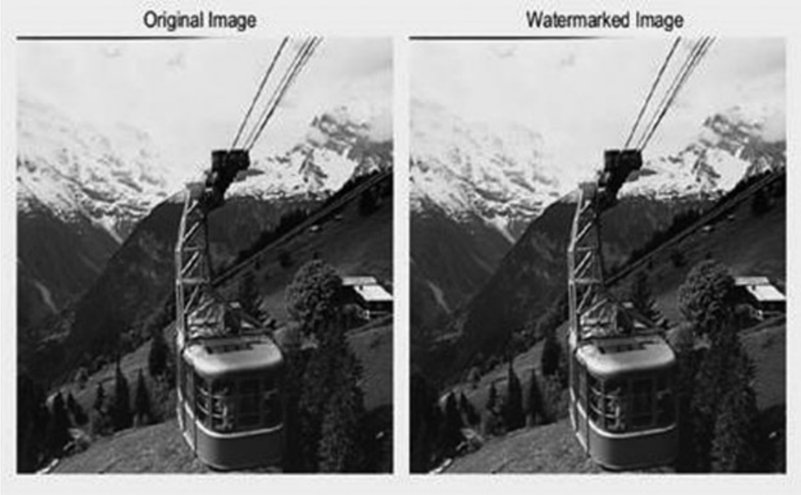

Figure 9: The original image and watermarked image of cablecar

Table 5: Experimental results of barbra.png

Secret watermarks	Embedding method	MSE	PSNR	SSIM	Insertion time (s)	Retrieval time (s)
W1	E1	0.00033	82.92101	1	0.539	0.136
	E2	0.00137	76.75317	1	0.550	0.140
	E3	0.00093	78.42454	1	0.611	0.151
	E4	0.00097	78.23380	1	0.590	0.149
W2	E1	0.00059	80.41288	1	0.565	0.140
	E2	0.00230	74.50583	1	0.584	0.146
	E3	0.00142	76.58748	1	0.656	0.155
	E4	0.00159	76.09406	1	0.624	0.150
W3	E1	0.00116	77.48746	1	0.623	0.143
	E2	0.00428	71.80853	1	0.669	0.149
	E3	0.00299	73.37304	1	0.693	0.163
	E4	0.00286	73.55402	1	0.682	0.160
W4	E1	0.00170	75.80342	1	0.667	0.149
	E2	0.00723	69.53781	1	0.692	0.153
	E3	0.00561	70.63417	1	0.721	0.169
	E4	0.00480	71.31594	1	0.714	0.165

“Tiffany” image failed to evade this attack; the boundary pixels of the largest blob covered the border pixels of the image. Therefore, cropping any part of the image would lead to lack the embedded watermarks. In addition, “cablecar” and “barbra” images succeeded in preserving the embedded watermarks into the original images.

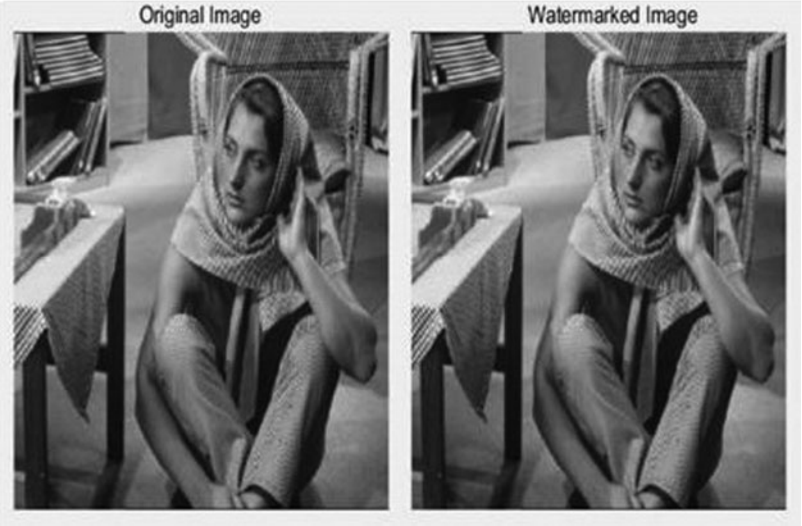

Figure 10: The original image and watermarked image of barbra

Table 6: Summary of attacks against watermarked images

Image	Embedding	W1	W2	W3	W4
Tiffany	E1	Fail	Fail	Fail	Fail
	E2	Fail	Fail	Fail	Fail
	E3	Fail	Fail	Fail	Fail
	E4	Fail	Fail	Fail	Fail
Cablecar	E1	Pass	Pass	Pass	Pass
	E2	Pass	Pass	Pass	Pass
	E3	Pass	Pass	Pass	Pass
	E4	Pass	Pass	Pass	Pass
Barbra	E1	Pass	Pass	Pass	Pass
	E2	Pass	Pass	Pass	Pass
	E3	Pass	Pass	Pass	Pass
	E4	Pass	Pass	Pass	Pass

On the other hand, we have tested the robustness of the algorithm against traditional LSB extraction algorithm to make sure if it will discover the watermark in the boundary pixels or not. It is implemented with the four different embedding methods, where the embedded data does not appear.

4.4 Upload & Download Process into Social Media

The last part of experiments is done as follows: The watermarked images are uploaded into a two social media sites. The first website called “Tumblr”, which is a popular site interested about images. In addition, it allows the users to share their images with the others. Therefore, it is a suitable environment for the experiments. Then, these images are downloaded back to test the presence of embedded watermarks or not. These results show that the embedded watermarks are retrieved successfully after the images are downloaded back from the site.

Based on the results, all watermarked images with different embedding methods passed the experiments, which will preserve the ownership of these images.

In addition, the second website used is twitter, which is one of the most popular social media platforms that allows users to broadcast images and short posts called tweets. The results shows that the watermarked images cannot pass the experiments of twitter website. The reason for that is the compression function used by twitter site. This compression can modify the pixels values of the image. A study to handle the compressed images should be done as a future work. Tab. 7 summarizes the results of these experiments.

Table 7: Summary of the presence of the watermarks

Social site	Embedding	W1	W2	W3	W4
Tumblr	E1	Pass	Pass	Pass	Pass
	E2	Pass	Pass	Pass	Pass
	E3	Pass	Pass	Pass	Pass
	E4	Pass	Pass	Pass	Pass
Twitter	E1	Fail	Fail	Fail	Fail
	E2	Fail	Fail	Fail	Fail
	E3	Fail	Fail	Fail	Fail
	E4	Fail	Fail	Fail	Fail

5 Conclusions

This paper proposed a technique to increase the security level of the uploaded images in the social media platforms while preserving the visual appearance of the images. The introduced method depends on extracting the largest blob of a digital image, then using the boundary pixels of this blob for hiding the watermarks. However, more than a blob can be used for the embedding process. The embedding process has been done by using four different methods. In addition, different experiments are carried out to evaluate the performance of the proposed method. For real test, the watermarked images are uploaded into a social media platform and downloaded back. The results show that the hidden watermarks cannot be extracted.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] T. Aichner, M. Grünfelder, O. Maurer and D. Jegeni, “Twenty-five years of social media: A review of social media applications and definitions from 1994 to 2019,” in *Cyberpsychology, Behavior, and Social Networking*, vol. 24. New Rochelle, NY: Mary Ann Liebert, Inc., pp. 215–222, 2021. http://doi.org/10.1089/cyber.2020.0134.

[2] H. Tankovska, “Number of social network users worldwide from 2017 to 2025,” 2021. [Online]. Available: https://www.statista.com.

[3] M. A. F. b. Jeffry and H. K. Mammi, “A study on image security in social media using digital watermarking with metadata,” in *Proc. AINS*, Miri, Sarawak, Malaysia, pp. 118–123, 2017.

[4] K. Gurpreet and K. Kaur, “Image watermarking using LSB (least significant bit),” *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 4, pp. 858–861, 2013.

[5] A. S. Kapse, S. Belokar, Y. Gorde, R. Rane and S. Yewtkar, “Digital image security using digital watermarking,” *International Research Journal of Engineering and Technology*, vol. 5, no. 3, pp. 163–166, 2018.

[6] M. Pavani, S. Naganjaneyulu and C. Nagaraju, “A Survey on LSB based steganography methods,” *International Journal of Engineering and Computer Science*, vol. 2, no. 8, pp. 2464–2467, 2013.

[7] T. K. Araghi, A. B. A. Manaf, M. Zamani and S. K. Araghi, “Taxonomy and performance evaluation of feature based extraction techniques in digital image watermarking,” *International Journal of Advances in Image Processing Techniques*, vol. 3, no. 1, pp. 20–23, 2016.

[8] A. Bamatraf, R. Ibrahim and M. N. B. M. Sallah, “Digital watermarking algorithm using LSB,” in *Proc. ICCAIE*, Kuala Lumpur, Malaysia, pp. 155–159, 2010.

[9] N. Jain, S. Meshram and S. Dubey, “Image steganography using LSB and edge-detection technique,” *International Journal of Soft Computing and Engineering*, vol. 2, no. 3, pp. 217–222, 2012.

[10] K. Joshi and R. Yadav, “A new method of image steganography using last three bit plane of gray scale images,” *Indian Journal of Science and Technology*, vol. 38, pp. 1–8, 2017.

[11] A. A. Attaby, M. F. M. M. Ahmed and A. K. Alsammak, “Data hiding inside JPEG images with high resistance to steganalysis using a novel technique: DCT-M3,” *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 1965–1974, 2018.

[12] S. Rahman, F. Masood, W. U. Khan, N. Ullah, F. Q. Khan *et al.,* “A novel approach of image steganography for secure communication based on lsb substitution technique,” *Computers, Materials & Continua*, vol. 64, no. 1, pp. 31–61, 2020.

[13] X. Liu, Y. Wang, J. Du, S. Liao, J. Lou *et al.,* “Robust hybrid image watermarking scheme based on KAZE features and IWT-SVD,” *Multimedia Tools and Applications*, vol. 78, no. 5, pp. 6355–6384, 2019.

[14] T. B. Moeslund, “Blob analysis,” in *Introduction to Video and Image Processing*, 1st ed., vol. 1. London: Springer-Verlag London, pp. 112–113, 2012.

[15] R. C. Gonzalez and R. E. Woods, “Digital image fundamentals,” in *Digital Image Processing*, 2nd ed., vol. 1. Upper saddle river, New Jersey, USA: Prentice Hall, pp. 60–66, 2002.

[16] A. Kaspers, “Blob detection,” Master’s thesis. University of Utrecht, Netherlands, 2011.

[17] B. M. ElHalawany, H. M. Abdel-Kader, A. TagEldeen, A. E. S. Ahmed and Z. B. Nossair, “Vision-based obstacles detection for a mobile robot,” in *Proc. INFOS*, Giza, Egypt, pp. 93–99, 2012.

[18] B. M. ElHalawany, H. M. Abdel-Kader, A. TagEldeen, A. E. Elsayed and Z. B. Nossair, “Modified A* algorithm for safer mobile robot navigation,” in *Proc. ICMIC*, Cairo, Egypt, pp. 74–78, 2013.

[19] U. Sara, M. Akter and M. S. Uddin, “Image quality assessment through FSIM, SSIM, MSE and PSNR—A comparative study,” *Journal of Computer and Communications*, vol. 7, no. 3, pp. 8–18, 2019.