Digital watermarking algorithm using LSB

Conference Paper · January 2011

DOI: 10.1109/ICCAIE.2010.5735066 · Source: IEEE Xplore

3 authors:

Abdullah Bamatraf
Universiti Teknologi Malaysia
4 PUBLICATIONS **104** CITATIONS
SEE PROFILE

Rosziati Ibrahim
Universiti Tun Hussein Onn Malaysia
118 PUBLICATIONS **544** CITATIONS
SEE PROFILE

Mohd Najib B. Mohd Salleh
Universiti Tun Hussein Onn Malaysia
79 PUBLICATIONS **635** CITATIONS
SEE PROFILE

Digital Watermarking Algorithm Using LSB

Abdullah Bamatraf
Information Technology and Multimedia Faculty
Tun Hussein Onn University, Malaysia
abdom45@hotmail.com

Rosziati Ibrahim
Information Technology and Multimedia Faculty
Tun Hussein Onn University, Malaysia
rosziati@uthm.edu.my

Mohd. Najib B. Mohd Salleh
Information Technology and Multimedia Faculty
Tun Hussein Onn University, Malaysia
najib@uthm.edu.my

Abstract__ In this paper, a simple and robust watermarking algorithm is presented by using the third and the fourth least significant bits (LSB) technique. The proposed algorithm is more robust than the traditional LSB technique in hiding the data inside the image. Using the proposed algorithm, we will embed two bits in the third and fourth LSB. Experimental results show that the quality of the watermarked image is higher.

Key Words: Digital watermarking, Grayscale images, secret data, LSB, PSNR.

1. INTRODUCTION

Illegal copying, modifying, tampering and copyright protection have become very important issues with the rapid use of internet [7]. Hence, there is a strong need of developing the techniques to face all these problems. Digital watermarking [1] emerged as a solution for protecting the multimedia data. Digital Watermarking is the process of hiding or embedding an imperceptible signal (data) into the given signal (data). This imperceptible signal (data) is called watermark or metadata and the given signal (data) is called cover work. The watermark should be embedded into the cover work, so that it should be robust enough to survive not only the most common signal distortions, but also distortions caused by malicious attacks. This cover work can be an image, audio or a video file. A watermarking algorithm consists of two algorithms, an embedding and an extraction (or detection) algorithm.

The idea of watermarking first appeared hundreds of years ago [2]. Watermarking technology was used to mark information authenticity by many different means. Watermarking technology has been used in computer as well. Most of the work on computer watermarking technology was for embedding a watermark into images, audio, and video files.

Media watermarking research is a very active area and digital image watermarking became an interesting protection measure and got the attention of many researchers since the early 1990s [3].

The rest of this paper is organized as follows: Section 2 describes the related work and Section 3 discusses the proposed algorithm. Results and discussion is given in Section 4 and finally, conclusion will be presented in Section 5.

2. RELATED WORK

In this section a literature review of digital watermarks used for images is presented. It describes the previous work which had been done on digital watermarks, including the analysis of various watermarking schemes and their results.

Ersin Elbasi et al [4] embed the watermark in a tree structure in the Discrete Wavelet Transform domain. For watermark embedding, the two level DWT decomposition of an NxN gray scale image I is computed. The same PRN sequence is embedded into the DWT coefficients higher than a given threshold T1 in the LL2 and HH2 bands. The watermark is also embedded into the children of DWT coefficients. The original DWT coefficients are replaced by the modified DWT coefficients. The final step is to compute the inverse DWT to obtain the watermarked image I'. For watermark detection, the DWT of the watermarked and possibly attacked image I* is computed. All the DWT coefficients higher than a given threshold T2 in the LL2 and HH2 bands are selected. Then the sum Z of all attacked DWT coefficients multiplied by either the embedded watermark or other random PRN sequence is computed, divided by the length of the PRN sequence. The sum is also computed for the children of modified DWT coefficients. A predefined threshold T is chosen for LL2 and HH2 bands and the HH1 band. In each band, if Z exceeds T, the conclusion is that the watermark is present.

Gil-Je Lee et al [5], presented a simple and robust watermarking scheme by using random mapping function. The idea of the proposed algorithm is watermark embedding which can be more robust than the traditional

LSB technique. Using the proposed algorithm, it makes the secure random coordinate of cover image to increase the robustness of the watermarked image.

Saeid Fazli et al [6], investigated trade-off between imperceptibility and robustness of LSB watermarking. In this algorithm significant bit-planes of the watermark image are put instead of lower bit-planes of the asset picture. So, they investigate the effect of image compression on the watermark, and finally they evaluate the robustness and imperceptibility by measuring the distortion due to watermarking using two quality metrics: MSE and 1 – SSIM.

Gaurav Bhatnagar et al [7] presented a new semi-blind reference watermarking scheme based on discrete wavelet transform (DWT) and singular value decomposition (SVD) for copyright protection and authenticity. We are using a gray scale logo image as watermark instead of randomly generated Gaussian noise type watermark. For watermark embedding, the original image is transformed into wavelet domain and a reference sub-image is formed using directive contrast and wavelet coefficients. We embed watermark into reference image by modifying the singular values of reference image using the singular values of the watermark. A reliable watermark extraction scheme is developed for the extraction of watermark from distorted image.

2.1 REVIEW OF LSB

The least significant bit (LSB) technique is used for simple operation to embed information in a cover image. The LSB technique is that inside of a cover image pixels are changed by bits of the secret message. Although the number was embedded into the first 8 bytes of the grid, the 1 to 4 least bits needed to be changed according to the embedded message. On the average, only half of the bits in an image will need to be modified to hide a secret message using a cover image. Because the quality of the Watermarked image is low, less than over the 4-bit LSB, changing the LSB of a pixel results in small changes in the intensity of the colors. These changes cannot be perceived by the human visibility system. However, a passive attacker can easily extract the changed bits, since, it has performed very simple operation. For example, Figure 1 shows the 1-bit LSB. In Figure 1, the pixel value of the cover image is $141(10001101)_2$ and the secret data is 0. It applies to LSB-1 that the changed pixel value of the cover is $140(10001100)_2$. LSB can store 1-bit in each pixel. If the cover image size is 256 x 256 pixel image, it can thus store a total amount of 65,536 bits or 8,192 bytes of embedded data.

1	0	0	0	1	1	0	**1**

Pixel value

0	0	1

Secret Data

1	0	0	0	1	1	0	**0**

Change Pixel Value

Figure 1. An example of 1 bit LSB

3. PROPOSED METHOD

Based on LSB technique, we propose a new watermarking algorithm. Most of researchers has proposed the first LSB but our proposed watermarking algorithm is using the third and fourth LSB for hiding the data. This is because of the security reason. So, no one will expect that the hidden data in the third and the forth LSB. Figure 2 shows the framework of the proposed method. First, we select the image which is a grayscale image and we will transfer the data to binary value after typing it. Then, we hide the data in the image using the proposed algorithm. Figure 3 shows the embedding algorithm in MATLAB. Then, we will get the watermarked image. Then, the receiver will retrieve the data back. Figure 4 shows the extracting algorithm in MATLAB. The data will be extracted from the watermarked image.

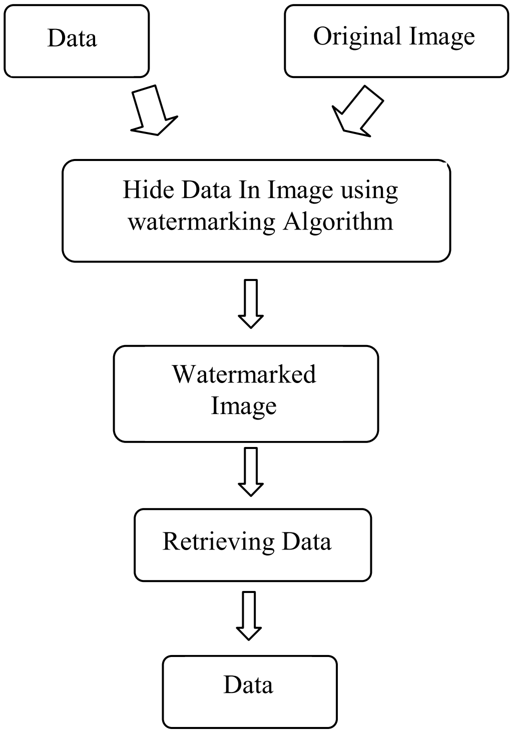


Figure 2. The framework of the proposed method

3.1 EMBEDDING ALGORITHM

In this section, we describe the embedding algorithm. After we select the image and type the secret data, we transfer the secret data to binary values and determine the coordinates of the image which the data will be embedded in. First, we will embed the length of the data in five pixels starting from the first coordinate which we select and jump by 5 until we embed it in the five pixels in the 3rd and 4th LSB, but if the length of data is more than 1023 characters, it will ask us to rewrite the data and it should be not more 1023 characters. Then, the data will be embedded in the image in the 3rd and 4th LSB. Then, watermarked image will be produced and it will be saved. Figure 3 shows the embedding algorithm.

```
B= Read the image
Type the secret message:
D=transfer the secret data to double values;
[m n]=size(B)
Coordinate y=200;
Coordinate x=1;
LM=the length of (D);
while LM>1023
  It will appear a message (rewrite the secret
    message which is supposed to be less than
    1024 characters)if LM>1023
  Type the secret message:
  D= transfer the secret data to double;
  LM=the length of (D);
end
w=transfer the double values (D) to binary
values;
LMbin=transfer the length of (D) from double to
binary;
for i=1:2:10
    put the value of (LMbin(i)) in the fourth LSB
    in(B(y,x))
    put the value of (LMbin(i+1)) in the third
    LSB in(B(y,x))
    x=x+5;
end
for i = 1 : LM
  for j=1 :2:8
    if x>m
      y=y+5;
      x=1;
    end
        put the value of (w(i,j)) in the fourth LSB
        in(B(y,x))
        put the value of (w(i,j+1)) in the third LSB
        in(B(y,x))
    x = x+5;
  end
end
imwrite(B,'watermarked_image.bmp','BMP')
```

Figure3. Embedding Algorithm

3.2 EXTRACTING ALGORITHM

In this section, we will describe the extracting algorithm. After receiving the watermarked image, we will get the length of the secret data from the 3rd and 4th LSB in the five pixels starting from the determined coordinates and jump by 5 until we get it from the five pixels. Then, we will get secret data also from the 3rd and 4th LSB in binary values. After that, we transfer the binary values to characters which will be shown as the secret data. Figure 4 shows the extracting algorithm.

```
B=imread('watermarked_image.bmp');
LMbin=we make a vector for the length of the
secret data contain from 10 elements;
Coordinate y=200;
Coordinate x=1;
for i=1:2:10
    put the value of the fourth LSB in(B(y,x))
    in(LMbin(i));
    put the value of the third LSB in(B(y,x))
    in(LMbin(i+1));
  x=x+5;
end
d=Transfer LMbin from binary to double;
wb=we make zeros matrix for the secret data
contain from 8 columns and LMbin rows;
for i = 1 : LMbin
   for j=1 :2:8
          Put the value of the fourth LSB in(B(y,x))
          in(wb(i,j));
          Put the value of the third LSB in(B(y,x))
          in(wb(i,j+1));
     x = x+5;
   end
end
c= Transfer wb from binary to double;
ws=Transfer c from double to char;
```

Figure4. Extracting Algorithm

4. EXPERIMENTAL RESULTS AND DISCUSSION

In our experimental results, four 512x512 grayscale images which are shown in Figure 5 were used as cover images. Once, we embed the same secret data which contain from 128 bytes in determined pixels in the forth and the third LSB and then, we got the watermarked images without noticeable distortion and subtract the watermarked image from the original image to see the difference between them. The second time, we embed the same secret data which contain from 1023 bytes in the four images and also we also got watermarked images without noticeable distortion on them and subtract the watermarked image from the original image to see the difference between them. Figure 6 shows the watermarked images and the difference between the original and the watermarked images. When we look to the difference between the original image and the watermarked image, we will see black image because the change in the 3rd and 4th LSB. The values of the 3rd and 4th LSB are 4 and 8. So, the maximum difference of the pixels between the two images will be 12 and the value 12 in grayscale images is nearly black as you see in Figure 6.

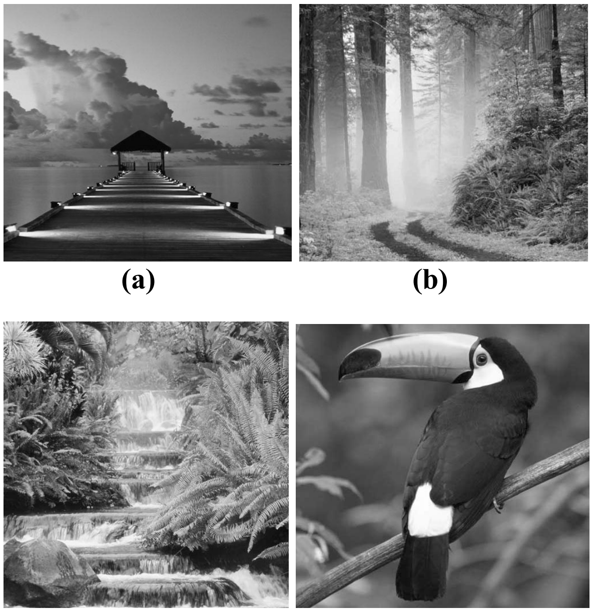
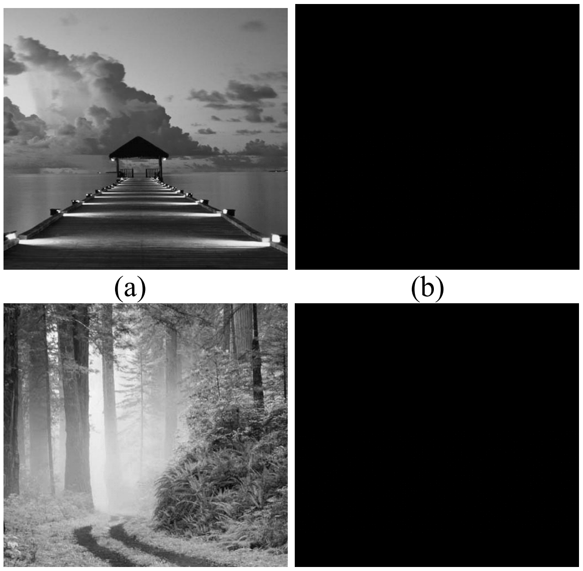


Figure 5: The four cover images: (a) Dock (b) Forest (c) Waterfall (d) Toco Toucan



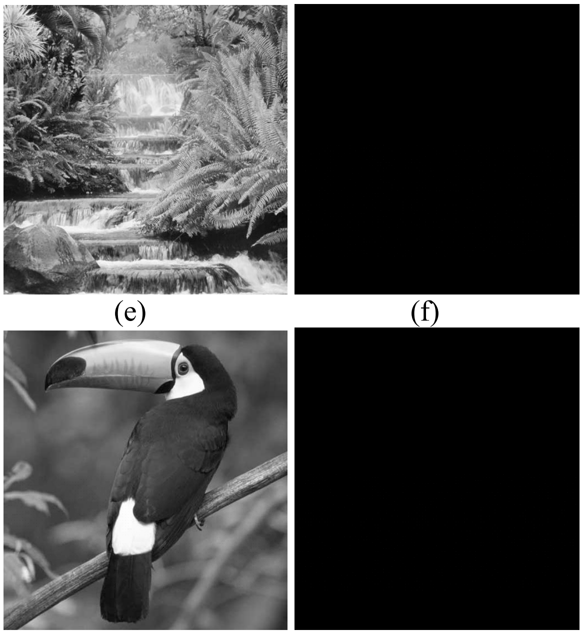


Figure 6: The four watermarked images and the difference: (a) Watermarked Dock (b) Difference of Dock (c) Watermarked Forest (d) Difference of Forest (e) Watermarked Waterfall (f) Difference of Waterfall (g) Watermarked Toco Toucan (h) Difference of Toco Toucan.

Notice that, there is no difference between the original and watermarked images. No distortion occurs for watermarked images. We got good result and we calculate the Peak signal-to-noise ratio (PSNR). The PSNR value was used to evaluate the quality of the watermarked images. The phrase peak signal-to-noise ratio (PSNR) is most commonly used as a measure of quality of reconstruction in image compression [4]. It is the most easily defined via the Mean Squared Error (MSE) which for two mXn images I and K where one of the images is considered as a noisy approximation of the other. MSE is defined as the following equation (2) and the PSNR is defined in equation (1).

$$\begin{aligned} PSNR &= 10 \cdot \log_{10}\left(\frac{MAX_I^2}{MSE}\right) \\ &= 20 \cdot \log_{10}\left(\frac{MAX_I}{\sqrt{MSE}}\right) \end{aligned} \qquad (1)$$

where MAX is equal to 255 in grayscale images, and MSE is the mean square error, which is defined as:

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

where I is the original image and K is the watermarked image.

Based on equations (1) and (2), we calculate the PSNR for our proposed algorithm to see the quality of the watermarked images. Table 1 shows the results of the PSNR calculated.

Image	PSNR for 128 bytes embedded	PSNR for 1023 bytes embedded
Dock	61.8427	52.7970
Forest	61.1210	52.5255
Waterfall	61.7931	52.6988
Toco Toucan	61.7138	52.5255

Table 1: PSNR of the watermarked images

Typical values for the PSNR are between 30dB and 40dB [4]. If the PSNR of the watermarked image is more than 30, it is hard to be aware of the differences with the cover image by the human eyes system. The cover images are shown in Figure 5, the watermarked images and the difference between them and the original images are shown in Figure 6. As they are shown, the invisibility of the watermark is good quality and the original image and the watermarked image cannot be distinguished by human visibility system (HVS). We calculate the PSNR and the result is shown in table 1. The result of PSNR of the four images are more than 52 when we embed 1023 byte as a secret data and if we embed less secret data we will get better PSNR as table 1 shown the PSNR is more than 61 when we embedded 128 bytes.

5. CONCLUSION

This paper proposed a new LSB based digital watermarking scheme with the fourth and third LSB in the grayscale image. After we have embedded the secret data in the third and fourth LSB in the image in determine coordinates, we got watermarked image without noticeable distortion on it. Therefore, this digital watermarking algorithm can be used to hide data inside image.

ACKNOWLEDGMENT

This research is supported by Fundamental Research Grant Scheme (FRGS) Vote 0738.

REFERENCES

[1] I.J. Cox, M.L. Miller, J.A. Bloom, Digital watermarking, Morgan Kaufmann, 2001.

[2] Mohannad Ahmad AbdulAziz Al-Dharrab," Benchmarking Framework for Software Watermarking" King Fahd University of Petroleum and Minerals, Dhahran, Saudi Arabia, June 2005.

[3] J. Nagra, C. Thomborson, and C. Collberg, (2002), A functional taxonomy for software watermark- ing, in M. Oudshoorn, ed., `Proc. 25th Australasian Computer Science Conference 2002', ACS, pp. 177-186.

[4] Ersin Elbasi and Ahmet M. Eskicioglu," A SEMI-BLIND WATERMARKING SCHEME FOR IMAGES USING A TREE STRUCTURE", Sarnoff Symposium, 2006 IEEE

[5] Saeid Fazli and Gholamreza Khodaverdi, "Trade-off between Imperceptibility and Robustness of LSB Watermarking using SSIM Quality Metrics", 978-0-7695-3944-7/10 $26.00 © 2010 IEEE DOI 10.1109/ICMV.2009.68

[6] Gil-Je Lee, Eun-Jun Yoon, Kee-Young Yoo, "A new LSB based DigitalWatermarking Scheme with Random Mapping Function", 978-0-7695-3427-5/08 $25.00 © 2008 IEEE DOI 10.1109/UMC.2008.33

[7] Gaurav Bhatnagar, Balasubramanian Raman," A new robust reference watermarking scheme based on DWT-SVD", 0920-5489/$ – see front matter © 2008 Elsevier B.V. All rights reserved. doi:10.1016/j.csi.2008.09.031.