Robust Lossless Image Data Hiding

Zhicheng Ni[1], Yun Q. Shi[1], Nirwan Ansari[1], Wei Su[1], Qibin Sun[2] and Xiao Lin[2]
1. New Jersey Institute of Technology, U.S.A
2. Institute for Infocomm Research, Singapore
shi@njit.edu

Abstract

Recently, among various data hiding techniques, a new subset, lossless data hiding, has drawn tremendous interests. Most of the existing lossless data hiding algorithms are, however, fragile in the sense that they will be defeated when compression or other small alteration is applied to the marked image. De Vleeschouwer et al's method is the only existing semi-fragile lossless data hiding technique (also referred to as robust lossless data hiding in this paper), which is robust against high quality JPEG compression. In this paper, we first pointed out that this technique has a fatal problem: salt-pepper noise caused by using modulo 256 addition. We then propose a novel robust lossless data hiding technique, which does not generate salt-pepper noise. This technique has been successfully applied to many commonly used images (including medical images, more than 1000 images in CorelDRAW database and JPEG2000 test images), thus demonstrating its generality. The experimental results show that the visual quality, payload and robustness are acceptable. In addition to medical and law enforcement fields, it has been applied to authenticate losslessly compressed JPEG2000 images.

1. Introduction

Recently, among various data hiding techniques, a new subset called lossless data hiding has drawn tremendous attention. Many techniques have been proposed, such as in [1-6]. However, most of them are *fragile* in the sense that the hidden data cannot be recovered when compression or other small alteration is applied to the marked image. Thus far, De Vleeschouwer et al's method [7] is the only existing robust lossless data hiding technique against high quality JPEG compression. This technique can be applied for semi-fragile authentication. That is, on the one hand, if the marked image does not change at all, the hidden data can be extracted out, and the original image can be recovered exactly, and hence it is authentic. On the other hand, if the marked image goes through compression to some extent, the hidden data can still be correctly extracted for semi-fragile authentication. Semi-fragile authentication may be more practical than fragile authentication since it allows some incidental modification, say, compression. The main idea of their algorithm comes from the patchwork theory. That is, each bit of the message is associated with a group of pixels, e.g., a block in an image. Each group is equally divided into two pseudo-random sets of pixels, i.e., zones A and B. The histogram of each zone is mapped to a circle (positions on the circle are indexed by the corresponding luminance, and the weight of the position is the number of pixels assuming this luminance). It is observed that in most cases the vectors pointing to the center of mass of zones A and B, as shown in Figure 1, are similar to each other. Hence slight rotation of these vectors in two opposite ways allows embedding one bit of information. As to the pixel values, rotations of the vectors correspond to luminance shifts. A diagram of embedding is shown in Figure 1.

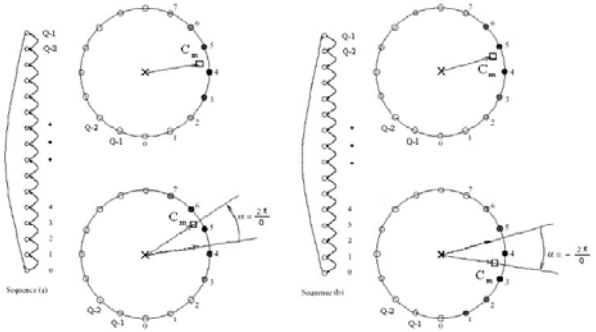

Figure 1: Data embedding diagram

From Figure 1, it is noted that modulo 256 addition is used to achieve losslessness. Therefore, this algorithm generates salt-pepper noise.

The followings are two pairs of figures demonstrating severe salt-pepper noise. Table 1 and Table 2 summarize the performance of [7] applied to medical images and JPEG2000 test images. There *robustness (bpp)* means the bit rate in bpp (bits per pixel) above which the hidden data can be retrieved with no error. Another drawback is that the marked image does not have high enough PSNR. According to our extensive tests, the PSNR of marked image is between 31 to 32 dB (with 1.5 k bits embedded into a gray image of 512x768). When the salt-pepper noise becomes severe, the PSNR may drop to below 20 dB. Therefore, our conclusion is that the lossless data hiding algorithms based on modulo 256 addition are not acceptable for practical usage. Thus, a new robust lossless data hiding technologies that can avoid the above mentioned drawbacks is called for.

0-7803-8603-5/04/$20.00 ©2004 IEEE.

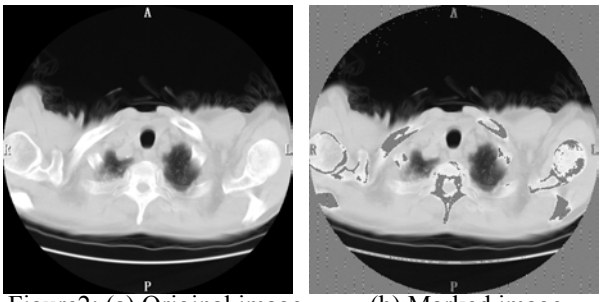

Figure2: (a) Original image (b) Marked image

Figure 3: (a) Original image (b) Marked image

Table 1: Test results for eight medical images

Images (512x512)	PSNR of marked image (dB)	Data embedding capacity (bits)	Robustness (bpp)
Mpic1	9.28	476	1.0
Mpic2	4.73	476	2.0
Mpic3	26.38	476	0.8
Mpic4	26.49	476	0.6
Mpic5	26.49	476	0.6
Mpic6	5.60	476	1.6
Mpic7	9.64	476	0.8
Mpic8	5.93	476	2.8

Table 2: Test results for eight JPEG2000 test images

Images (1536x1920)	PSNR of marked image (dB)	Data embedding capacity (bits)	Robustness (bpp)
N1A	17.73	1410	0.8
N2A	17.73	1410	2.2
N3A	23.73	1410	0.6
N4A	19.67	1410	1.2
N5A	17.28	1410	1.2
N6A	23.99	805	0.6
N7A	20.66	1410	1.4
N8A	14.32	805	1.4

2. A novel robust lossless image data hiding algorithm

In order to be robust against JPEG/JPEG2000 compression, we should select a robust parameter to embed data. In this proposed algorithm, a statistic quantity is selected as the parameter. Below is the illustration of this idea.

2.1. Main idea

For a given 8×8 image block, we split it into two sub-sets A and B as shown in Figure 4, i.e., subset A consists of all pixels marked by '+', the other B '-'. Each sub-set has 32 pixels.

+	-	+	-	+	-	+	-
-	+	-	+	-	+	-	+
+	-	+	-	+	-	+	-
-	+	-	+	-	+	-	+
+	-	+	-	+	-	+	-
-	+	-	+	-	+	-	+
+	-	+	-	+	-	+	-
-	+	-	+	-	+	-	+

Figure 4: Difference pair pattern

For each block, we calculate the difference value α which is defined as the arithmetic average of differences of pixel pairs within the block. We may choose a pair as two horizontal neighboring pixels. Below is the formula. In this example, n is equal to 32.

$$\alpha = \frac{1}{n}\sum_{i=1}^{n}(a_i - b_i)$$

Since the pixel values in a local block are highly correlated and have spatial redundancy, the difference value α is expected to be very close to zero. The experimental results have supported this observation.

The distribution of the difference value α of many image blocks is shown in the Figure 5. Note that most values of α are very close to zero (or the mean value of this distribution is zero).

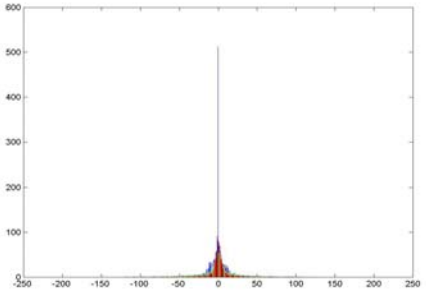

Figure 5: The distribution of the difference value α.

Since the difference value α is based on the statistics of all pixels in the block, this value α has certain robustness against attacks (such as JPEG/JPEG2000 compression and other slight alteration). We select this difference value α as a robust quantity for embedding information bit.

2.2. Bit embedding strategy

Case 1: the difference value α is located within a defined threshold. If 1 is to be embedded, we shift the difference value α to the right side or left side beyond a threshold, by adding or subtracting a fixed number from each pixel value within one subset, such as subset A, as shown in Figure 6. If 0 is to be embedded, the block is intact.

Case 2: the difference value α is located outside the threshold. No matter bit 1 or 0 is to be embedded, we always embed bit 1, thus shifting the value α further away beyond the threshold. We then rely on error correction code (ECC) to correct the bit error introduced in this case.

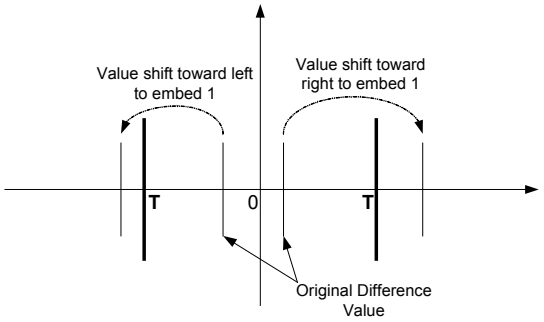

Figure 6: Embedding a bit '1'

2.3. Over/underflow problem

In some cases, the pixel values in a block are very close to the ends of histogram, such as 0 or 255 in the 8-bit case. The modification of the pixel values may lead to over/underflow problem, which means the modified pixel values are beyond the range of [0,255]. Instead of using modulo 256 addition, we propose a new technique to solve this issue. That is, if the pixel values only fall into one side of the histogram, we may shift the pixel value towards the other side to avoid the over/underflow problem. In the worst case, if there are some pixel values with the block, which are close to the both sides, respectively we do nothing to the block, which means we actually embed bit 0 to that block no matter the actual bit to be embedded is 1 or 0. The introduced error bit will be corrected by using ECC.

2.4. Error correction code and chaotic mixing

As discussed above, in order to handle different bit embedding situations, it is unavoidable to introduce some erroneous bits. To losslessly recover the hidden data and the original image, error correction codes (such as BCH code) are applied to correct the erroneous bits.

To combat the burst error, which may fail our algorithm, we introduce chaotic mixing [8] on the watermark matrix to spread the burst error evenly in the whole watermark matrix so that ECC can work effectively.

2.5. Data extraction

Data extraction is actually the reverse process of data embedding. For a given marked image, we first split it into non-overlapping blocks and then calculate the difference value α for each block in the same way as that in data embedding.

If the difference value α is outside the threshold, then bit 1 is extracted and the difference value is shifted back, meaning that the pixel value of one sub-set is back to its original value. If the difference value α is within the threshold, then bit 0 is extracted and nothing is done on the pixel value of that block. In this way, we can extract the watermark and obtain the original image without any distortion.

3. Experimental results

We have successfully applied our proposed algorithm to some commonly used grayscale images such as 'lena', 'baboon', etc., some medical images, more than 1000 images in CorelDraw image database, and JPEG2000 color test images. Note that salt-pepper noise is not generated at all since we do not use modulo 256 addition in our algorithm. The embedding capacity is above 1024 or 512 bits and is adjustable. (Note that this is often sufficient for authentication purpose). The average PSNR is above 38 dB. The tested images can resist the JPEG2000 compression attack from 2.0 bpp to 0.2 bpp.

Figure 7 and 8 are two pairs of test images. Note that no visible artifacts exist, indicating a significant performance improvement has been achieved as compared with [7].

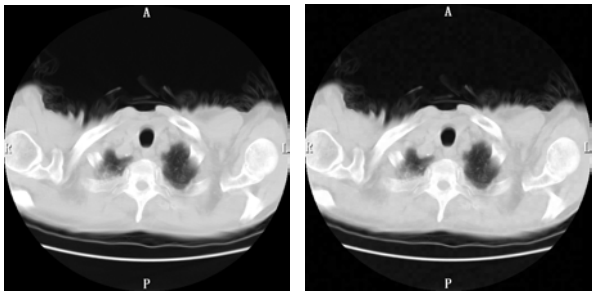

Figure 7: Medical image (a) Original (b) Marked

Figure 8: JPEG2000 test image (a)Original (b)Marked

Tables 3 and 4 summarize test results for eight medical images and eight JPEG2000 test images, respectively.

Table 3: Test results for eight medical images

Images (512x512)	PSNR of marked image (dB)	Data embedding capacity (bits)	Robustness (bpp)
Mpic1	40.4	768	0.8
Mpic2	40.8	560	0.8
Mpic3	40.3	792	0.6
Mpic4	40.3	792	1
Mpic5	40.3	792	0.8
Mpic6	40.7	560	0.8
Mpic7	40.4	768	0.4
Mpic8	40.6	560	0.8

Table 4: Test results for eight JPEG2000 test images

Images (1536x1920)	PSNR of marked image (dB)	Data embedding capacity (bits)	Robustness (bpp)
N1A	45.1	1398	0.8
N2A	43.1	1398	1.6
N3A	45.1	1398	1
N4A	45.2	1398	1
N5A	45.5	1200	1
N6A	45.0	1267	0.4
N7A	40.6	1398	1.2
N8A	41.5	798	1.4

4. Conclusion

We have proposed a novel robust lossless image data hiding technique, which employs a statistical quantity as a parameter for data embedding, thus successfully avoiding salt-pepper noise. This technique has a few advantages over the existing robust lossless data hiding technique: 1) no salt-and-pepper noise; 2) applicable to many commonly used images (including medical images, more than 1000 images in CorelDRAW database and JPEG2000 test images); 3) average PSNR of marked images is above 39 dB; 4) robust to JPEG/JPEG2000 compression to a certain extent; 5) data embedding capacity is above 1024 bits or 512 bits (often sufficient for authentication purpose and the capacity can be adjusted according to requirement).

This proposed scheme has been included in a proposal [9] on unified authentication framework to JPSEC (Part 8 of JPEG) recently, in which the proposed technique is used to authenticate losslessly compressed JPEG2000 images.

ACKNOWLEDGMENT

The work has been supported in part by NJCST via NJWINS, and by the Digital Data Embedding Technologies group of the Air Force Research Laboratory, Rome Research Site, Information Directorate, Rome NY, under contract F30602-03-1-0264. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation there on. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies, either expressed or implied, of Air Force Research Laboratory, or the U.S. Government.

References

[1] C. W. Honsinger, P. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data," US Patent: 6,278,791, 2001.

[2] M. Goljan, J. Fridrich, and R. Du, "Distortion-free data embedding," *Proceedings of 4th Information Hiding Workshop*, pp. 27-41, Pittsburgh, PA, April 2001.

[3] G. Xuan, J. Zhu, J. Chen, Yun Q. Shi, Z. Ni, W. Su "Distortionless Data Hiding Based on Integer Wavelet Transform," *IEE, Electronics Letters*, Volume 38, No 25, pp.1646-1648, Dec.2002

[4] Z. Ni, Yun Q. Shi, N. Ansari and W. Su, "Reversible Data Hiding," *IEEE International Symposium on Circuits and Systems,* Bangkok, Thailand, May 2003.

[5] M. Celik, G. Sharma, A.M. Tekalp, E. Saber, "Reversible data hiding," in *Proceedings of the International Conference on Image Processing 2002*, pp. 157-160, Rochester, NY, September 2002.

[6] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890-896, August 2003.

[7] C. De Vleeschouwer, J. F. Delaigle and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Tran. Multimedia,* vol. 5, pp. 97-105, March 2003.

[8] G. Voyatzis and I. Pitas, "Chaotic mixing of digital images and applications to watermarking," in *European Conference of Multimedia Applications, Services Techniques (ECMAST'96*), 2, pp. 687-695, May 1996.

[9] Z. Zhang, G. Qiu, Q. Sun, X. Lin, Z. Ni and Y. Q. Shi, "A unified authentication framework for JPEG2000 images," WG1N2946, JPEG Strasbourg meeting, July 2003; WG1N3107, JPEG Hawaii meeting, December 2003.