EAI Endorsed Transactions
on Internet of Things Research Article EAI.EU

Improved JPEG image watermarking in data compression domain using block selection strategy

M. Zairi [1,*], T. Boujiha [1] and A. Ouelli [1]

[1] Engineering Science Laboratory Ibn Tofail University, ENSA – B.P 241 Kenitra, Morocco.

Abstract

Nowadays, the joint photographic experts group (JPEG) is still the most dominant format among various digital image formats used in our daily life. Therefore, watermarking JPEG images is important and useful for many applications such as archive management and image authentication. Along with the embedding capacity and visual quality that must be considered for uncompressed images, the storage size of the watermarked JPEG image should be also considered. In this paper, based on the philosophy behind the JPEG encoder and the statistical properties of discrete cosine transform (DCT) coefficients, we propose a new JPEG image watermarking in the data compression domain, in which only coefficients with values 1 and -1 are expanded to carry message bits, coefficients with values 2 and -2 are shifted, and all the others coefficients remain unchanged. Moreover, a block selection strategy based on the number of 1 and -1 coefficients in each 8×8 block is proposed, which can be utilized to adaptively choose DCT coefficients for data embedding. Experimental results demonstrate that by using the proposed method, we can easily realize high embedding capacity and good visual quality. Furthermore, the storage size increase for the watermarked JPEG image is well lessened.

Keywords: JPEG, image watermarking, block selection, histogram shifting.

Received on 11 January 2021, accepted on 28 January 2021, published on 08 February 2021

doi: 10.4108/eai.8-2-2021.168690

1. Introduction

Image Watermarking is the process of embedding a hidden stream of bits in a digital image. Digital watermarking has many applications such as broadcast monitoring, device control, owner identification, proof of ownership, transaction tracking, content authentication, and copy control [1]. The main specifications of a watermarking system are Robustness, Imperceptibility, and Capacity. In fact, there is a trade-off between these factors; importance of each depends on the application.

The algorithms found in the literature can be divided according to the domain of insertion into three categories:

- Watermarking in Spatial Domain
- Watermarking in Spectral Domain
- Watermarking in Data Compression Domain

Algorithms belonging to the last category have recently become of great interest. Secret data embedded in the compressed image would not be easily found and people would not suspect that there is something important hidden in the image.

Furthermore, among various digital image formats used in our daily life, the joint photographic experts group (JPEG) is the most popular. JPEG standard is one of the oldest and most commonly used digital image formats. Most current media broadcasting corporations and digital devices use JPEG image compression to store information in graphic form. Data hiding in a JPEG compression domain in a reversible manner is useful for image archive management, image authentication, and image privacy.

As JPEG image format is the most used compressed format, many related algorithms have been developed for hiding information in JPEG images. Upham [2] proposed a famous hiding tool for JPEG images named Jpeg-Jsteg, where the hidden data are embedded into the least significant bits (LSBs) of the quantized DCT coefficients whose values are not 0, 1 or -1. Fridrich et al. [3] developed a new idea to losslessly compress the LSB plane of some selected JPEG mode coefficients to make space for reversible data embedding. Chang et al. [4] proposed a lossless

[*] Corresponding author. Email: mouradzairi@gmail.com

steganography scheme to hide secret data in the quantized DCT coefficients of each block in JPEG images.

Many algorithms for JPEG images watermarking have been proposed in the past few years. However, improvements regarding the embedding capacity, visual quality, and file size preservation have still been a field of research. In this paper, some new insights on how to select DCT coefficients for data hiding will be analysed first. Then, on the basis of these analyses, we will present a new scheme for JPEG images watermarking. The main difference between our method and the previously proposed ones are as follows:

(i) The secret message bits are only embedded in AC coefficients with values 1 and -1.
(ii) AC coefficients with values 2 and -2 are shifted.
(iii) All other coefficients remain unchanged in the embedding process.
(iv) A novel block selection strategy based on the number of AC coefficients with values 1 and -1 in each 8×8 block is proposed.

Experimental results demonstrate that by using the proposed scheme, high embedding capacity and good visual quality can be easily obtained. Meanwhile, the storage size of the host JPEG file is well preserved. The remainder of this paper is organized as follows. In Section 2, the proposed JPEG image watermarking in the data compression domain is introduced. Experimental results and comparison with previous works are discussed in Section 3. Finally, we conclude in Section 4.

2. Proposed Scheme

In this section, an overview of JPEG compression is discussed first, and then, we introduce the proposed JPEG image watermarking in the data compression domain. In subsection 2.2, the embedding of the AC coefficients will be explained. Subsection 2.3 explains the block selection method. Subsection 2.4 explains the recovery of the original DCT coefficients and the extraction of the payload.

2.1. Overview of JPEG Compression

JPEG Compression is an algorithm developed by the Joint Photographic Experts Group with the aim of minimizing the file size of photographic image files. The key steps of the JPEG compression process [5, 6], are shown in Fig. 1. The JPEG legacy encoder consists of three parts: discrete cosine transform (DCT), quantization step, and entropy encoding. The original image is divided into non-overlapping 8×8 pixel blocks, subtracted by 127 for normalization, transformed using the two-dimensional DCT function, and entropy coding is applied using the Huffman table. In the quantization step, DCT coefficients are divided according to the quantization table and rounded to the nearest integer as follows:

$$D(u,v) = \text{IntegerRound}\left(\frac{F(u,v)}{Q(u,v)}\right) \qquad (1)$$

where F(u,v) represents the original DCT coefficient, Q(u,v) represents the corresponding value in the quantization table, and D(u,v) represents the quantized DCT coefficient.

JPEG standard does not specify the DCT quantization table, but it provides a recommended table in their Annex. This means that anyone can use their own quantization table by scaling the table elements using different quality factors [5, 6]. When a different quantization table is used, different image qualities and compression ratios are achieved. The recommended quantization table is shown in Table 1. Quality factor (QF) is a value ranging from 1 to 100, and scaled quantization table Qs is obtained using the following equation:

$$SF = \begin{cases} \frac{5000}{QF} & if\ QF < 50 \\ 200 - (2\ \times QF) & if\ QF \geq 50 \end{cases} \qquad (2)$$

$$\mathrm{Q_s(i,j) = max}\left\{\left[\mathrm{Q(i,j)} \times \frac{\mathrm{SF}}{100}\right], 1\right\} \qquad (3)$$

where SF is the scale factor, Q is the recommended quantization table value, and [x] represents the round operator of x.

The quantized DCT coefficients are arranged in a zigzag scanning order, the very first coefficient in a quantized coefficient vector is referred to as the direct current (DC) coefficient, whereas the 63 others are referred to as the alternating current (AC) coefficients.

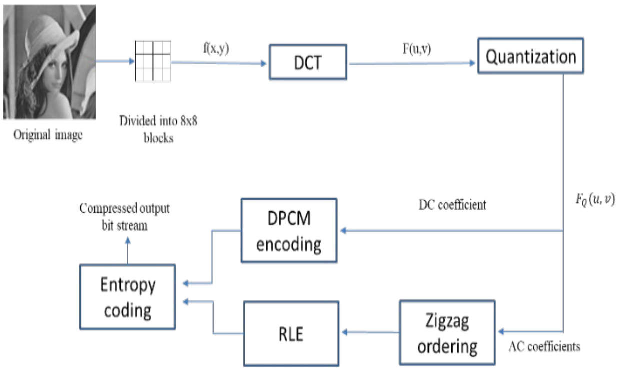
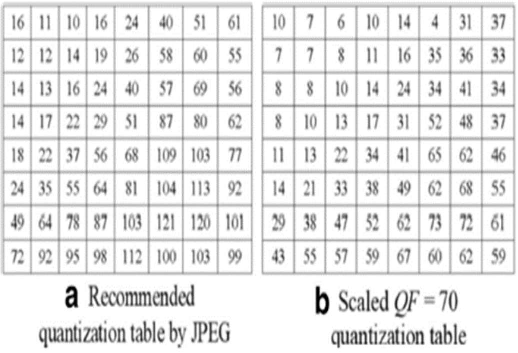


Figure 1. Block diagram of the legacy JPEG compression.

Table 1. EXAMPLES OF JPEG QUANTIZATION TABLES



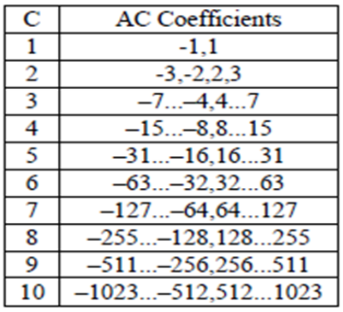
16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

a Recommended quantization table by JPEG

10	7	6	10	14	4	31	37
7	7	8	11	16	35	36	33
8	8	10	14	24	34	41	34
8	10	13	17	31	52	48	37
11	13	22	34	41	65	62	46
14	21	33	38	49	62	68	55
29	38	47	52	62	73	72	61
43	55	57	59	67	60	62	59

b Scaled *QF* = 70 quantization table

Table 2. CATEGORIES ASSIGNED TO COEFFICIENT VALUES



C	AC Coefficients
1	-1,1
2	-3,-2,2,3
3	–7...–4,4...7
4	–15...–8,8...15
5	–31...–16,16...31
6	–63...–32,32...63
7	–127...–64,64...127
8	–255...–128,128...255
9	–511...–256,256...511
10	–1023...–512,512...1023

The AC coefficients are encoded in a specific RLE format as intermediate symbols (R/C,V). Here, R (0 ≤R≤15) denotes the zero run length (i.e., the number of zero AC coefficients before the next nonzero AC coefficient), and C (1 ≤C≤ 10) denotes the category of the next nonzero AC coefficients (i.e., the number of bits needed to represent the amplitude of the next nonzero AC coefficient), which is shown in Table 2. There are 160 combinations of R and C. Each R/C is encoded with a variable-length code (VLC) from the Huffman table; the code length ranges from 1 to 16. In addition, because the run length of zero coefficients may exceed 15, the value R/C = F0H is defined to represent a run length of 15 zero coefficients followed by a coefficient of zero amplitude (this can be interpreted as a run length of 16 zero coefficients), and a special value R/C = 00H is used to code the end-of-block (EOB) when all remaining coefficients in the block are zero, where “H” implies that “F0” and “00” are hexadecimal numbers. Thus, there are in total 162 different VLC codes in the standard Huffman table.

Note that all R/C values are encoded with variable-length Huffman codes, but the V values are not Huffman coded. Each V value is encoded with a variable-length integer (VLI) code, whose length in bits is given in the first column of Table 2. They are appended to the Huffman-coded R/C to form the final JPEG bitstream.

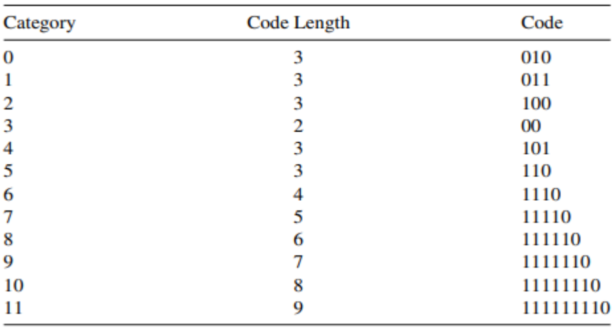
The DC coefficients are pre-compressed differently using the differential pulse code modulation (DPCM), they are coded as (Size,Value) pair. The code for the Size is derived from Table 3 and the code for the Value is simply its binary code. The DC component value in each 8x8 block is large and varies across blocks but is often close to that in the previous block. The DPCM encode the difference between the current and previous 8x8 block as follows:

$$e_{dc}(j) = DC(Jj) - DC(j-1) \quad (4)$$

where e_{dc} is the DC differential value.

Finally, the symbol string is Huffman coded to obtain the final compressed bitstream. After pre-pending the header, we obtain the final JPEG file. For more detail, please refer to the JPEG guidelines released by International Telecommunication Union [7].

Table 3. JPEG Luma default Huffman table for DC coefficients



Category	Code Length	Code
0	3	010
1	3	011
2	3	100
3	2	00
4	3	101
5	3	110
6	4	1110
7	5	11110
8	6	111110
9	7	1111110
10	8	11111110
11	9	111111110

2.2. Selection of Coefficients for Embedding

The very first coefficient in a quantized coefficient block is referred to as the DC coefficient, whereas 63 others are referred to as AC coefficients. The proposed method will only embed the payload in the AC coefficients valued either +1 or -1. For each AC coefficient C, the following method is used to embed the payload bits b ϵ 0,1 and creates the watermarked AC coefficient C′:

$$\mathrm{C}' = \begin{cases} C + sign(C) * b & if\ |C| = 1 \\ C + sign(C) & if\ |C|\ =\ 2 \\ C & else \end{cases} \quad (5)$$

where

$$sign(x) = \begin{cases} 1 & if\ x > 0 \\ 0 & if\ x = 0 \\ -1 & if\ x < 0 \end{cases}$$

A bit b is embedded if and only if C is either +1 or -1. If C is equal to +2 or -2, it is shifted by +1 and -1, respectively. Notice that the others coefficients including the zero coefficients are not modified as explained in the beginning of this section. Before we continue with the rest of the explanation, the following histogram shifting terms are summarized for clarity:

- Embeddable coefficients: AC coefficients valued either +1 or −1.
- Shiftable coefficients: AC coefficients which are greater than +2 or less than −2.
- Unchangeable coefficients: All others AC coefficients.

2.3. Block selection strategy

We proposed two block selection strategies, the first is based on the number of AC coefficients valued either +1 or −1, the second is based on the number of 0 AC coefficients. A block with a higher number of those coefficients will be at the top with the highest embedding priority and a block with less number of those coefficients will be at the bottom with lowest embedding priority. Using this statistical feature, embedding only in +1 and -1 first in selected blocks effectively reduce distortion in the watermarked image. Additionally, the file size increase is also lessened. The modification of others AC coefficient increases the file size. This is because whenever an AC coefficient is modified, an extra symbol is needed to be coded. Therefore, the proposed method leads to smaller distortion and smaller file size increase.

2.4. Extraction and image recovery

In this proposed scheme, the payload should be extracted and the original host image should be recovered correctly without any errors. The extraction and recovery are done simultaneously.

The message or the watermark extraction and the host image restoration can be described as:

$$\tilde{b} = \begin{cases} 0 & if\ |C'| = 1 \\ 1 & if\ |C'| = 2 \end{cases} \qquad (6)$$

$$\tilde{C} = \begin{cases} 0 & if\ |C'| = 1 \\ sign(C') & if\ |C'|\ =\ 2 \\ 2 * sign(C') & if\ |C'|\ =\ 3 \\ C' & else \end{cases} \qquad (7)$$

where b˜ and C˜ represent the extracted message bit and the recovered AC coefficient respectively.

3. Experimental results

In our experiments, we have used 16 popular images from the popular USC-SIPI database including the Lena, Baboon, airplane (F-16), and House. These files are compressed using the JPEG standards with optimized Huffman table. The proposed method was evaluated by comparing the visual quality and file size increase. The PSNR is used to evaluate the visual quality of the watermarked JPEG image, and it is calculated between the original JPEG image and the watermarked JPEG image. File size increase is measured by calculating the number of bytes.

3.1. Visual Quality

For evaluating visual quality, the PSNR is calculated between the watermarked JPEG image and the original JPEG image. Table 4 shows the numerical results of PSNR values on the listed four images applying a quality factor of 70 and a payload size ranging from 8000 bits to 20000 bits. The complete PSNR results for the four images with payload size ranging from 2000 bits to 20000 bits are shown in Fig. 2. Fig. 3 shows the four watermarked JPEG image compressed with a factor quality of 70 and watermarked by a secret data with a payload size of 16000 bits. The experiment shows that the proposed method produces a good visual quality.

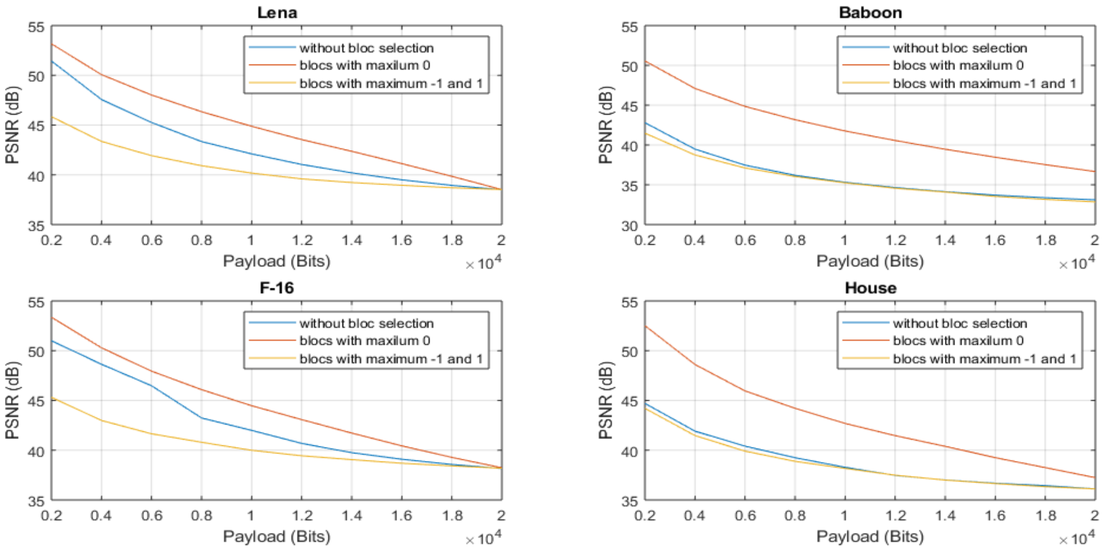


Figure 2. PSNR corresponding to different embedding payloads with and without block selection strategy

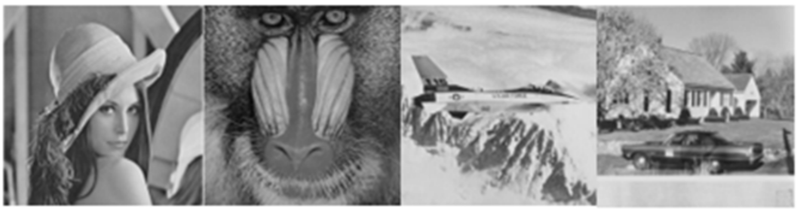


Figure 3. Watermarked JPEG images: Lena, Baboon, F-16, and House (from left to right)

Table 4. PSNR corresponding to different embedding payloads for the four test images

Test images	*Payload (Bits)*	*without block selection strategy*	*blocks with maximum of 0*	*blocks with maximum of -1 and 1*
Lena	8000	43,34	46,35	40,92
	12000	41,04	43,54	39,59
	16000	39,50	41,13	38,95
	20000	38,50	38,51	38,53
Baboon	8000	36,20	43,17	36,04
	12000	34,65	40,56	34,57
	16000	33,70	38,46	33,54
	20000	33,11	36,66	32,85
F16 Airplane	8000	43,24	46,09	40,80
	12000	40,69	43,08	39,44
	16000	39,09	40,44	38,69
	20000	38,16	38,23	38,18
House	8000	39,25	44,21	38,88
	12000	37,47	41,46	37,49
	16000	36,68	39,25	36,65
	20000	36,11	37,24	36,11

Table 5. Comparison in terms of PSNR (dB) with payload size of 8000 bits

Image	*Methods*			
	[10]	*[9]*	*[8]*	*Proposed*
F16	40,59	40,41	44,24	46,09
Lena	41,37	41,34	45,24	46,35
Baboon	38,03	37,95	42,49	43,17
Boat	40,41	39,74	43,43	44,38
Peppers	42,63	41,99	45,49	46,31
Average	40,61	40,29	44,18	45,42

Table 6 Comparison in terms of PSNR (dB) with payload size of 16,000 bits

Image	*Methods*			
	[10]	*[9]*	*[8]*	*Proposed*
F16	37,43	37,33	38,26	40,44
Lena	38,48	38,29	39,71	41,13
Baboon	34,39	34,28	36,95	38,46
Boat	37,14	36,8	38,89	40,22
Peppers	39,33	38,83	40,69	41,89
Average	37,35	37,11	38,90	40,43

The proposed method was furthermore evaluated by comparing the visual quality against the three state-of-the-art schemes [8, 9, 10]. The experiment shows that when we select the blocks with maximum 0 values the proposed method has the highest PSNR than all the previous works, but as the payload increases, the block selection has no more effect because all embeddable AC coefficients are used as we can note for Lena and F-16 images in Table 4 and Figure 2 for a payload size of 20000 bits.

Table 7. Comparison in terms of PSNR (dB) on USC-SIPI 9 images with payload size of 10,000 bits

Image	*Methods*			
	[10]	*[9]*	*[8]*	*Proposed*
4.2.01	42,65	42,33	43,9	44,82
4.2.03	37,05	36,97	40,69	41,93
4.2.05	39,68	39,36	42,71	44,47
4.2.07	41,25	41,07	43,89	44,99
5.2.08	39,38	39,21	41,47	43,03
5.2.10	38,64	38,52	39,9	42,01
7.1.01	41,02	40,82	42,62	44,06
Boat	39,01	38,78	41,92	43,20
House	38,23	38,23	40,7	42,69
Average	39,66	39,48	41,98	43,47

3.2. *File Size Preservation*

It is obvious that it is important to consider the file size preservation along with visual quality in JPEG image watermarking. The file size of the watermarked images obtained by the proposed method is on average less when we select the blocks with maximum values of -1 and 1. Table 8 shows the file size increase for each watermarked JPEG image using payload sizes ranging from 8000 bits to 20000 bits. Fig. 4 shows that embedding only in 1 and -1 AC coefficients and shifting only AC coefficients valued 2 or -2 reduce the file size increase of the marked JPEG image.

Table 8. File size increase corresponding to different embedding payloads for the four test images

Test images	*Payload size (Bits)*	*without block selection strategy*	*blocks with maximum of 0*	*blocks with maximum of -1 and 1*
Lena	8000	1349	1368	1210
	12000	1960	1999	1836
	16000	2597	2602	2490
	20000	3213	3199	3218
Baboon	8000	1265	1322	1164
	12000	1895	1927	1752
	16000	2522	2542	2364
	20000	3151	3165	2962
F16 Airplane	8000	1257	1303	1074
	12000	1837	1876	1649
	16000	2417	2444	2287
	20000	2973	2939	2949
House	8000	1226	1331	1118
	12000	1824	1922	1705
	16000	2411	2511	2299
	20000	3010	3099	2943

Similarly, the proposed method was evaluated by comparing the preservation of the image size against the three state-of-the-art schemes [8, 9, 10]. The experiment shows that when we select the blocks with maximum values of 1 and -1, the proposed method has the highest size preservation than all the previous works, but as the payload increases, the block selection has no more effect because all embeddable AC coefficients are used.

Table 9. Comparison in terms of file size increase (Bytes) with payload size of 8,000 bits

Image	*Methods*			
	[10]	*[9]*	*[8]*	*Proposed*
F16	1377	1372	1169	1074
Lena	1375	1397	1305	1210
Baboon	1339	1375	1286	1166
Boat	1258	1315	1250	1195
Peppers	1378	1433	1400	1154
Average	1345	1378	1282	1160

Table 10. Comparison in terms of file size increase (Bytes) on USC-SIPI 9 images with payload size of 10,000 bits

Image	*Methods*			
	[10]	*[9]*	*[8]*	*Proposed*
4.2.01	1674	1714	1656	1461
4.2.03	1501	1521	1563	1466
4.2.05	1574	1666	1444	1357
4.2.07	1672	1712	1667	1460
5.2.08	1506	1524	1495	1394
5.2.10	1453	1451	1712	1509
7.1.01	1290	1475	1491	1326
Boat513	1564	1592	1588	1483
House	1629	1602	1580	1401
Average	1540	1584	1577	1429

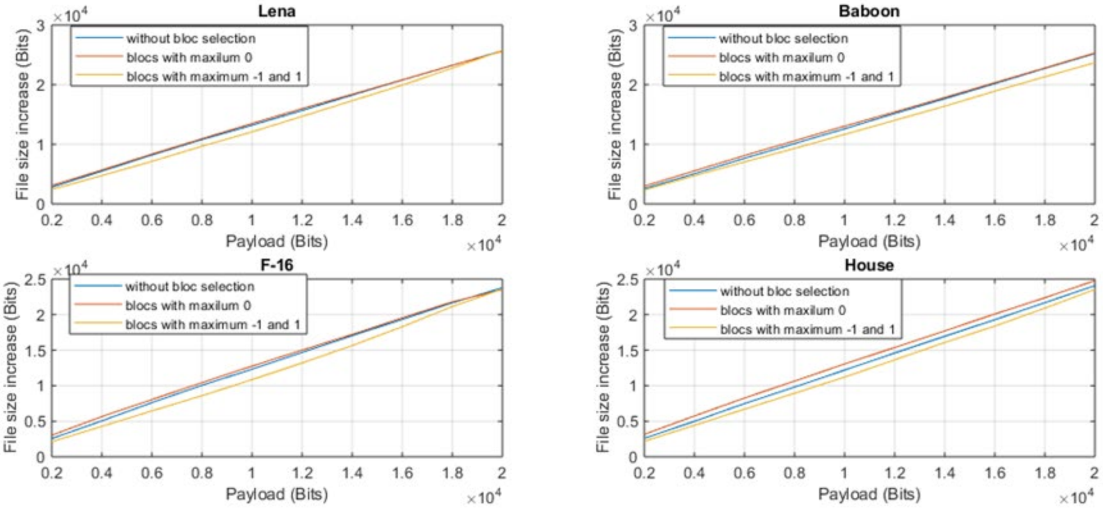


Figure 4. File size increase corresponding to different embedding payloads with and without block selection strategy

In the present work, the process of watermarking an image includes embedding some AC coefficients and shifting others which lead inevitably to increase its initial size and reduce its visual quality. Block selection strategies significantly improve the visual quality and reduce the increase of the file; by selecting the blocks with maximum values of -1 and 1, we got more AC coefficients to embed in, less coefficients to shift, and few needed blocks to embed the entire hidden message which result finally in enhancing the size preservation as shown in Tables 9 and 10. In contrast, selecting blocks with maximum 0 values which mean blocks with minimum values of -1 and 1 leads to less embeddable AC coefficients in each block and the distortion will be distributed over more blocks which results in enhancing the visual quality as shown in Tables 5, 6, and 7. In fact, there is a trade-off between these two factors; importance of each depends on the application.

4. Conclusion

A new watermarking technique for JPEG images is of great significance due to its large series of applications. However, any embedding in JPEG images introduces inevitably more distortion and increases its initial size. In this paper, we present a new scheme for JPEG images watermarking. The main contributions of this paper are as follows: 1) On the basis of the philosophy behind the encoder and the distribution of quantized DCT coefficients, a new DCT coefficient modification-based method is proposed. In the proposed method, only AC coefficients with values 1 and -1 are expanded to carry the message bits and only AC coefficients with values 2 and -2 are shifted. 2) A novel block selection strategy has been proposed in this paper, which may result in better visual quality and less storage size of the marked JPEG file. 3)

The proposed scheme has strong practicability because high embedding capacity and good visual quality can be easily obtained. Meanwhile, the storage size increase of the watermarked JPEG image can be well lessened. Experimental results demonstrate that our proposed method can achieve better performance both in visual quality and file size preservation compared to the previous works in [8-13]. The future work includes the extension of our proposed technique to other category and types of images, for example, color images, DICOM images, and JPEG 2000 which is a wavelet-based image compression standard [14-16].

References

[1] Cox, J.; Miller, M. L.; Bloom, J. A.; Fridrich J. & Kalker T., Digital Watermarking and Steganography, Morgan Kaufmann Pub., Elsevier Inc, 2008.

[2] D.Upham,1997,http://www.funet.fi/pub/crypt/steganography/jpeg-jsteg-v4.diff.gz.

[3] J. Fridrich, M. Goljan, R. Du, Lossless data embedding for all image formats, in: Proceedings of SPIE, Electronic Imaging 2002, Security and Watermarking of Multimedia Contents IV, San Jose, California, vol. 4675, pp. 572-583.

[4] C.C. Chang, C.C. Lin, C.S. Tseng, W.L. Tai, Reversible hiding in DCT-based compressed images, Information Sciences 177 (13), 2007, pp. 2768-2786.

[5] W. B. Pennebaker, and J. L. Mitchell, JPEG: Still Image Data Compression Standard, Springer, 1993

[6] G.K. Wallace, The JPEG still picture compression standard. IEEE Trans. Consumer Electron. 38(1),1992, pp. 30-44.

[7] Int. Telecommunication Union CCITT Recommendation T.81, Information Technology Digital Compression and Coding of Continuous-tone Still Images-Requirements and Guidelines 1992.

[8] F. Huang, X. Qu, H.J. Kim, Reversible data hiding in JPEG image. IEEE Trans.Circuits Syst. Video Technol. 26(9), 2016, pp1610–1621.

[9] G. Xuan, Y. Q. Shi, Z. Ni, P. Chai, X. Cui and X. Tong, Reversible data hiding for JPEG images based on histogram pairs, In Proceedings of the International Conference on Image Analysis and Recognition, Montreal,Canada, 2007, pp. 715–727.

[10] H. Sakai, M. Kuribayashi, and M. Morii, Adaptive reversible data hiding for JPEG images, in Proceedings of the International Symposium on Information Theory and Its Applications, Auckland, New Zealand,2008, pp. 1–6.

[11] Wedaj, F.T., Kim, S., Kim, H.J. et al. Improved reversible data hiding in JPEG images based on new coefficient selection strategy. J Image Video Proc.,2017, pp 63.

[12] Xie, X., Lin, C. & Chang, C. A reversible data hiding scheme for JPEG images by doubling small quantized AC coefficients. Multimed Tools Appl 78,2019, pp. 11443-11462.

[13] Y. Qiu, Z. Qian, H. He, H. Tian and X. Zhang, Optimized Lossless Data Hiding in JPEG Bitstream and Relay Transfer Based Extension, in IEEE Transactions on Circuits and Systems for Video Technology, doi: 10.1109/TCSVT.2020.

[14] M.D. Adams. The JPEG-2000 still image compression standard, ISO/IEC ITC I/SC 29/WG I, Document N2412, September 2001.

[15] Sadayasu Ono, Junji Suzuki. JPEG2000 Technology. Beijing: Science Press, April 2004.

[16] Hwai-Tsu Hu, Ling-Yuan Hsu, Exploring DWT–SVD–DCT feature parameters for robust multiple watermarking against JPEG and JPEG2000 compression, Computers & Electrical Engineering, Volume 41, 2015, pp 52-63.