ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) | 978-1-7281-7605-5/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICASSP39728.2021.9415118

REVERSIBLE DATA HIDING IN JPEG IMAGES FOR PRIVACY PROTECTION

*Yuxuan Huang, Xin Cao, Hao-Tian Wu**

School of Computer Science and Engineering,
South China University of Technology,
Guangzhou, GD 510006, China

Yiu-ming Cheung

Department of Computer Science,
Hong Kong Baptist University,
Kowloon Tong, Hong Kong, China

ABSTRACT

Recently, how to protect privacy in JPEG images has become an important issue in social networks. Besides encryption, reversible visual transformation has been studied for content and privacy protection. In this paper, an improved algorithm is proposed to conceal privacy information in JPEG images. By adopting face detection, the regions to be protected can be identified so that the DC and some AC coefficients in them are modified. The changes that have been made are reversibly saved in the other AC coefficients, and the block selection information is hidden into the whole image also by reversible data hiding. As a secret key is employed to control the process of transformation, the protected content can hardly be obtained without knowing the key. We conduct the proposed algorithm on a set of test images, and compare the performance with several existing algorithms in terms of information leakage, file size increment, and image quality. The experimental results show that improved image quality and less information leakage can be achieved with the proposed reversible transformation algorithm.

***Index Terms*—** JPEG image, face detection, privacy protection, shuffling, reversible data hiding

1. INTRODUCTION

JPEG standard [1] is widely used to save bandwidth and storage space due to high compression ratios. In the emerging social media, how to protect privacy in JPEG images has become an important issue. Recently, reversible visual transformation (RVT) has been studied to protect content and privacy in JPEG images but image quality needs to be improved.

The existing algorithms can be classified into two categories: (1) those operated on the luminance component only (e.g., [2, 3, 4]), (2) algorithms operated on all components (e.g., [5, 6, 7, 8]). For instance, the reversible data hiding (RDH) algorithm in [9] is applied in [2] to modify the AC coefficients for image quality degradation. Moreover, two successive AC coefficients are modified to mark the protected blocks, which makes the protection reversible. However, file size of the protected JPEG image increases and part of privacy may be leaked. In addition, the protected blocks can be easily identified and the AC coefficients may be restored by brute force attack [10]. In [3], block scrambling is combined with RDH to achieve invisibility of protected regions and preserve file size. But some protected content can still be identified and image quality may be affected because the chrominance component is neglected. In Yuan's algorithm [5] and Li's algorithm [7], image masking and data hiding (e.g., [11, 12, 13, 14, 15, 16, 17]) are used to cover human face, but the operations are lossy so that original images cannot be exactly recovered. The algorithm in [8] modifies the signs of the quantized discrete cosine transform (DCT) coefficients in the protected regions and inserts the location information in JPEG file header, which is referred to Scrambling JPEG.

In general, a desirable privacy protection algorithm for JPEG images should meet the following criteria. (1) privacy protection, i.e., it is hard to recognize the protected region, (2) reversibility, i.e., the original JPEG image can be obtained when needed, (3) security, i.e., only the authorized party can recover the original image and the privacy will not be revealed after a variety of attacks, (4) file size preservation, i.e., the file size of JPEG image should not be significantly changed by privacy protection. To achieve these goals, a new RVT algorithm is proposed for privacy protection in JPEG images. First, face detection is adopted to aid the user to select the regions to be protected. To blur the content to be protected, the DC and some AC coefficients in selected regions are modified according to their mean values, while the changes are saved in the other AC coefficients by using a RDH algorithm [18]. In addition, location information of the selected blocks is hidden into the whole image by adopting the RDH algorithm in [19]. A secret key is employed to control the process of RVT so that the original image can be recovered with the correct key. We conducted the proposed RVT algorithm on a set of test images, and the performance was compared with several existing algorithms in terms of information leakage, file size increment, and image quality. The experimental results show that improved image quality and less information leakage can be achieved with the new algorithm. In addition, our algorith-

*The contact author (E-mail: wuht@scut.edu.cn). This work was partly supported by NSFC (Nos.61772208, 61672444), and the SCUT Fundamental Research Funds for the Central Universities of China (x2js-D2190700).

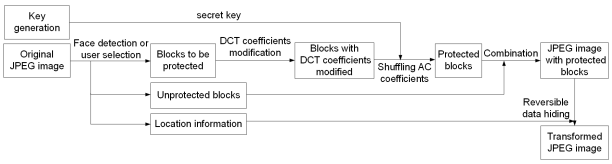

Fig. 1. A flowchart of transforming an original JPEG image.

m introduces only a little file size increment of JPEG images.

In the next section, the proposed algorithm is presented in details. The experimental results are given in Section 3. Finally, we draw concluding remarks in Section 4.

2. THE PROPOSED METHOD

The algorithm to be proposed is applicable to color JPEG images by conducting it on the luminance and chrominance components, respectively. For gray-scale JPEG images, the algorithm can be applied on the luminance component only. Two processes are included in the proposed algorithm, i.e., visual transformation and reverse transformation.

2.1. Transforming original JPEG image

A flowchart of transforming a JPEG image is illustrated in Fig. 1. First, the regions to be protected are selected with the aid of face detection. The locations of the selected blocks are recorded as a binary bitstream, while the low frequency coefficients in them are modified, including the luminance and chrominance components. The changes are reversibly hidden into the other AC coefficients with a RDH algorithm, and the AC coefficients in the selected regions are shuffled with a secret key known by the authorized users only. Finally, the location information is reversibly saved into the JPEG image.

2.1.1. Key generation

A secret key is used in the algorithm to generate the transformed image. However, critical information may be leaked if the same key is shared by the user. To enhance the security, a pseudo random number generator (PRNG) controlled with a secret key is utilized to generate a binary sequence to be used in the following steps. To be unique, the secret key is generated by calculating the SHA-256 hash value [20] of the user password and characteristics of the JPEG image.

2.1.2. Region selection

A face detection technique [21] is applied to identify the regions to be protected while user autonomous selection is also allowed. Since a JPEG image is divided into 8×8 blocks, the locations of the selected blocks are recorded. As each block is assigned with an index number, the index of the top left block, the height and width in blocks are stored with a binary stream with a fixed length. The length depends on the size of a protected JPEG image. When there are 256 blocks in a JPEG image on the scale of 16 blocks per row and 16 blocks per column, a block index, height and width of the protected region are represented by 8, 4 and 4 bits, respectively. The generated bit stream and its length are concatenated, which will be hidden in the last step of visual transformation.

2.1.3. Modifying DCT coefficients

In the proposed algorithm, both the luminance and chrominance components are modified to protect the privacy information. For simplicity of illustration, only the case of one region to be protected is introduced, while the case of multiple regions can be processed by repeating the procedure.

In the privacy preserving algorithms for JPEG images based on the DCT scrambling (e.g., [8, 22, 23]), the protected region will be irregularly deformed with block effects. To improve visual quality of the protected regions as well as achieve smaller file size, the first 6 DCT coefficients of the luminance component (including the DC and AC ones) in zigzag order are modified, which is also enough to protect the content of selected regions according to the experimental results. For each of them, a DCT coefficient in a selected block is modified by $d_j{}' = d_j - delta'$, where $d_j{}'$ represents the j^{th} ($j = 1, 2, 3, 4, 5, 6$) modified coefficient in the selected block, d_j represents the original one and $delta'$ is obtained as follows. Firstly, the average of the j^{th} DCT coefficient of all blocks in the selected region is calculated by

$$avg = \frac{\sum_{i=1}^{n} d_{ij}}{n}, \tag{1}$$

where n represents the number of blocks and i is the block index, which ranges from 1 to n. Then the difference between d_j and avg is calculated by $delta = d_j - avg$. After that, $delta$ is quantized to $delta'$ with the method used in [24] by applying Eq. (2) to reduce the embedded data, i.e.,

$$delta' = \begin{cases} 8 \times round(\frac{delta}{8}), & \text{if } delta \geq 0 \\ 8 \times \lfloor \frac{delta}{8} \rfloor + 4, & \text{if } delta < 0 \end{cases}, \tag{2}$$

where $round(\cdot)$ and $\lfloor \cdot \rfloor$ are the round and floor functions, respectively. The value of $\frac{|delta'|}{4}$ is converted to a binary stream and hidden into the AC coefficients in the same block with the RDH algorithm in [18]. The length of the bit stream is fixed to 9 bits, because each DCT coefficient ranges from -1023 to 1023 and $\frac{|delta'|}{4}$ ranges between 0 to 511, which can be represented by 9 bits. For recovery, $delta'$ can be obtained by multiplying $\frac{|delta'|}{4}$ with 4 and the sign can be judged from the parity of $\frac{|delta'|}{4}$ (i.e., odd or even). As totally 6 DCT coefficients are modified in each block, the 6^{th} to 59^{th} AC coefficients are used to carry the changes that have been made. For each AC coefficient a_j and a bit value s (0 or 1) to

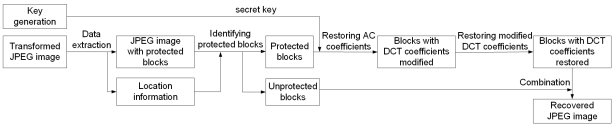

Fig. 2. A flowchart of recovering an original JPEG image.

be hidden, data embedding is conducted by $a_j{}' = a_j \times 2 + s$, where $a_j{}'$ $(j = 6, 7, 8, ..., 59)$ represents the j^{th} modified AC coefficient while a_j is the original one.

For the chrominance component, only DC coefficients in selected blocks are modified. Since human eyes are sensitive to luminance information but not to chrominance, the simplified operation on chrominance component helps to reduce the increment of file size while protecting the selected region.

2.1.4. Shuffling AC coefficients

Shuffling AC coefficients contains two parts, i.e., flipping the signs of AC coefficients in the selected blocks and then permutating AC coefficients in all components of a JPEG image. For each selected region, a random sequence denoted by R_1 is generated with a PRG, which takes a secret key as the seed. The signs of AC coefficients in a selected block are flipped according to R_1. Furthermore, the AC coefficients in each selected block are scrambled with another sequence R_2 by shuffling the run length code with the method in [25].

2.1.5. Hiding location information

After the selected blocks are modified, a histogram modification based algorithm proposed in [19] is applied to embed the location information. The operation is performed by

$$a' = \begin{cases} 0, & \text{if } a = 0 \\ a + i \times sign(a), & \text{if } a = 1 \text{ or } a = -1 \\ a + 1, & \text{if } a > 1 \\ a - 1, & \text{if } a < -1 \end{cases}, \quad (3)$$

where a' represents the modified AC coefficient, a is the original one, i is a bit value (0 or 1) to be embedded and $sign(\cdot)$ represents the sign function. The number of modified AC coefficients is determined by length of the location information. Suppose there are n bits to be hidden, the first n AC coefficients are chosen to be modified for RDH.

2.2. Recovering original JPEG image

A flowchart of recovering an original JPEG image is shown in Fig. 2, where a secret key is required. By extracting the location information from the protected JPEG image, the positions of the selected regions can be known. Then the shuffled AC coefficients are reordered according to the sequence generated with the secret key. After that, the modified DCT coefficients in each component and each block are restored so that the original JPEG image is recovered.

2.2.1. Extracting location information

According to Section 2.1.5, only the AC coefficients with value ± 1 or ± 2 contain the embedded data. Hence, a bit value i may be extracted from an AC coefficient a' by

$$i = \begin{cases} 1, & \text{if } a' = -2 \text{ or } a' = 2 \\ 0, & \text{if } a' = -1 \text{ or } a' = 1 \end{cases}. \quad (4)$$

Then the AC coefficient is restored by

$$a = \begin{cases} 0, & \text{if } a' = 0 \\ -1, & \text{if } a' = -1 \text{ or } a' = -2 \\ 1, & \text{if } a' = 1 \text{ or } a' = 2 \\ a' + 1, & \text{if } a' < -2 \\ a' - 1, & \text{if } a' > 2 \end{cases}, \quad (5)$$

where a is the restored AC coefficient. From the extracted data, the protected regions can be known so that the selected blocks are identified.

2.2.2. Recovering the shuffled AC coefficients

For each component in the protected region, the random sequence R_1 and R_2 used in visual transformation are generated by using the secret key as the seed of PRG. The permuted AC coefficients are reordered by restoring the shuffled run length code with R_2. Then the signs of all AC coefficients are flipped with R_1 to restore the modified AC coefficients.

2.2.3. Recovering the original DCT coefficients

Since the luminance and chrominance components are modified in visual transformation, the recovery procedure is conducted on both of them, respectively. For the luminance component in a protected region, the recovery is performed as follows. At first, the embedded data is extracted from the 6^{th} to 59^{th} AC coefficients by $s = a_j{}' \ mod \ 2$ and the original AC coefficient a_j is restored by $a_j = \lfloor \frac{a_j{}'}{2} \rfloor$, where s is an extracted bit value while $a_j{}'$ $(j = 6, 7, 8, ..., 59)$ is the j^{th} AC coefficient. With the extracted bits, $\frac{|delta'|}{4}$ can be obtained so that the difference $delta'$ is calculated by

$$delta' = \begin{cases} \frac{|delta'|}{4} \times 4, & \text{if } \frac{|delta'|}{4} \ mod \ 2 = 0 \\ -\frac{|delta'|}{4} \times 4, & \text{if } \frac{|delta'|}{4} \ mod \ 2 = 1 \end{cases}. \quad (6)$$

Finally, an original DCT coefficient d_j is obtained by $d_j = d_j{}' + delta'$, where $d_j{}'$ is the j^{th} modified DCT coefficient for $j = 1, 2, 3, 4, 5, 6$. For each protected block in the chrominance component, the recovery procedure is similar by recovering the DC coefficients only.

(a) By the proposed algorithm (b) By the algorithm in [3] (c) By the proposed algorithm (d) By the algorithm in [3]

Fig. 3. Examples of four transformed JPEG images.

Table 1. Performance comparisons with five algorithms.

Algorithm	PSNR(dB)	File size increment
Proposed	$+\infty$	1.15%
Niimi's algorithm [2]	$+\infty$	7.09%
Cao's algorithm [3]	$+\infty$	0.39%
Yuan's algorithm [5]	50.5926	10.80%
Li's algorithm [7]	49.7378	0.51%
Scrambling JPEG [8]	$+\infty$	2.04%

3. EXPERIMENTAL RESULTS

All test JPEG images were downloaded from the Images of Groups Dataset [26] and the experiments were implemented in the MATLAB environment. The experimental results include: (1) comparison with related work in terms of reversibility and file size increment, (2) security analysis.

3.1. Comparison with existing algorithms

One thousand test images were used and four protected images are shown in Fig. 3, where the proposed algorithm and the one in [3] have been applied, respectively. For comparison, only one region is selected to be protected in each JPEG image, though multiple regions may be selected in practise. Fig. 3 shows that better visual quality and less information leakage are achieved with the proposed algorithm. Specifically, Fig. 3(b) exposes the private information in the image, such as age, which is unrecognizable in Fig. 3(a). In addition, the average PSNR value between the protected region and the original one is 11.63 dB, so the confidentiality of the protected content is ensured. As shown in Table 1, the proposed algorithm is compared with other five algorithms in [2, 3, 5, 7, 8]. The average PSNR value between a restored test image and the original one, as well as the average file size increment after transformation, are shown in Table 1, respectively. It can be seen that Yuan's algorithm and Li's algorithm are lossy while the others are completely reversible. The proposed algorithm introduces 1.15% increment of JPEG file size, which is lower than most of the existing algorithms except [3].

3.2. Security analysis

We analyze the security by performing a brute force attack [10] and a differential cryptanalysis. Suppose that the attacker knows the transform process of the proposed algorithm and infer the location information without knowing the secret key. An exhaustive search of the possible seeds for PRG may be conducted. Since a SHA-256 hash value of 256 bits is used as the seed for shuffling AC coefficients, the number of the possible trials is 2^{256}. An exhaustive search of the possible combinations of flipped AC coefficients is more difficult. According to the work in [3], the block number ranges from 25 to 1600 in a protected region while most are within 25 to 400. Besides, the experiments in [22] show that there are 26 nonzero AC coefficients in a block on average. Given a protected region containing 25 blocks with 26 nonzero coefficients per block, the number of combinations to recover the flipped AC coefficients is $2^{25*26} = 2^{650}$, showing that the proposed algorithm is resistant to brute force attack. The number of pixels change rate (NPCR) is usually used to evaluate the ability of image cryptography's sensitivity to the input. According to [27], the expected value of NPCR is 99.6094%. we randomly chose 100 different coefficients (one at a time) in each original test image and changed their values slightly. Then the NPCR was computed for each case and the final result was 98.1057%, indicating that the proposed algorithm is resistant to differential cryptanalysis.

4. CONCLUSION

We have proposed a reversible privacy protection algorithm for JPEG images. The privacy content is concealed by modifying DCT coefficients and then shuffling AC coefficients. In addition, reversible data hiding is adopted while image quality of the unprotected regions is preserved. Besides user manual operations, face detection is adopted to aid the selection of the regions to be protected. The experimental results have shown that reversibility and less information leakage are achieved. Moreover, the proposed algorithm introduces little file size increment and it is resistant to brute force attack and differential cryptanalysis. Our future work is to improve image quality and extend the algorithms to JPEG2000 format [28].

5. REFERENCES

[1] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Transactions on Consumer Electronics*, vol. 38, no. 1, pp. xviii–xxxiv, 1992.

[2] M. Niimi, F. Masutani, and H. Noda, "Protection of privacy in JPEG files using reversible information hiding," in *2012 International Symposium on Intelligent Signal Processing and Communications Systems*. IEEE, 2012, pp. 441–446.

[3] X. Cao, Y. Huang, H. Wu, and Y. Cheung, "Content and privacy protection in JPEG images by reversible visual transformation," *Applied Sciences*, vol. 10, no. 19, pp. 6776, 2020.

[4] H. Wu, R. Jia, J. Dugelay, and J. He, "Reversible image visual transformation for privacy and content protection," *Multimedia Tools and Applications*, 2020, https://doi.org/10.1007/s11042-020-09985-1.

[5] L. Yuan and T. Ebrahimi, "Image transmorphing with JPEG," in *2015 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2015, pp. 3956–3960.

[6] S. Çiftçi, A. O. Akyüz, and T. Ebrahimi, "A reliable and reversible image privacy protection based on false colors," *IEEE transactions on Multimedia*, vol. 20, no. 1, pp. 68–81, 2018.

[7] W. Li, R. Ni, and Y. Zhao, "JPEG photo privacy-preserving algorithm based on sparse representation and data hiding," in *International Conference on Image and Graphics*. Springer, 2017, pp. 575–586.

[8] L. Yuan, P. Korshunov, and T. Ebrahimi, "Secure JPEG scrambling enabling privacy in photo sharing," in *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. IEEE, 2015, vol. 4, pp. 1–6.

[9] D. Coltuc, "Improved capacity reversible watermarking," in *2007 IEEE International Conference on Image Processing*. IEEE, 2007, vol. 3, pp. III–249.

[10] "Brute force," `http://sipi.usc.edu/database/`.

[11] A. Wstfeld, "F5-a steganographic algorithm high capacity despite better steganalysis," *Lecture notes in computer science*, vol. 2137, pp. 289–302, 2001.

[12] H. Wu and J. Huang, "Secure JPEG steganography by lsb+ matching and multi-band embedding," in *2011 18th IEEE International Conference on Image Processing*. IEEE, 2011, pp. 2737–2740.

[13] J. He, X. Pan, H. Wu, and S. Tang, "Improved block ordering and frequency selection for reversible data hiding in JPEG images," *Signal Processing*, vol. 175, pp. 107647, 2020.

[14] Y. Wang, R. Ni, and Y. Zhao, "A novel block sorting scheme for reversible data hiding in JPEG images," in *2018 14th IEEE International Conference on Signal Processing (ICSP)*, 2018, pp. 389–394.

[15] J. He, J. Chen, and S. Tang, "Reversible data hiding in JPEG images based on negative influence models," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 2121–2133, 2020.

[16] M. Xiao, X. Li, B. Ma, X. Zhang, and Y. Zhao, "Efficient reversible data hiding for JPEG images with multiple histograms modification," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 1–1, 2020.

[17] Z. Yin, Y. Ji, and B. Luo, "Reversible data hiding in JPEG images with multi-objective optimization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 8, pp. 2343–2352, 2020.

[18] H. Wu, H. Tang, and J. Wang, "A reversible visual transformation algorithm for JPEG images," *Journal of South China University of Technology (Natural Science)*, vol. 46, no. 5, pp. 16, 2018.

[19] F. Huang, X. Qu, H. J. Kim, and J. Huang, "Reversible data hiding in JPEG images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 26, no. 9, pp. 1610–1621, 2015.

[20] "Federal information processing standards publication 180-2, announcing thesecure hash standard," 2002.

[21] S. Yu, "Lib face detection," `https://github.com/ShiqiYu/libfacedetection`.

[22] N. Ruchaud and J. L. Dugelay, "JPEG-based scalable privacy protection and image data utility preservation," *IET Signal Processing*, vol. 12, no. 7, pp. 881–887, 2018.

[23] N. Ruchaud and J. Dugelay, "Privacy protecting, intelligibility preserving video surveillance," in *2016 IEEE International Conference on Multimedia & Expo Workshops (ICMEW)*, Los Alamitos, CA, USA, 2016, pp. 1–6, IEEE.

[24] D. Hou, W. Zhang, and N. Yu, "Image camouflage by reversible image transformation," *Journal of Visual Communication and Image Representation*, vol. 40, pp. 225–236, 2016.

[25] M. Takayama, K. Tanaka, A. Yoneyama, and Y. Nakajima, "A video scrambling scheme applicable to local region without data expansion," in *2006 IEEE International Conference on Multimedia and Expo*. IEEE, 2006, pp. 1349–1352.

[26] A. C. Gallagher and T. Chen, "Understanding groups of images of people," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 256–263.

[27] V. Patidar, N. K. Pareek, and K. K. Sud, "A new substitutioncdiffusion based image cipher using chaotic standard and logistic maps," *Communications in Nonlinear Science & Numerical Simulation*, vol. 14, no. 7, pp. 3056–3075, 2009.

[28] M. Rabbani, "JPEG2000: Image Compression Fundamentals, Standards and Practice," *Journal of Electronic Imaging*, vol. 11, no. 2, 2002.