Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

# Supplementary materials for

Yongning GUO, Guodong SU, Zhiqiang YAO, Wang ZHOU, 2024. Reversible data hiding scheme for encrypted JPEG bitstreams using adaptive RZL rotation. *Front Inform Technol Electron Eng*, 25(10):1353-1369. https://doi.org/10.1631/FITEE.2300749

## 1 Security analysis and comparison

In the proposed scheme, the encryption of DCs and ACCs is conducted to generate the encrypted JPEG bitstreams. The encryption of DC phase includes the row-wise shuffling and reversing of DC coefficients and the grouping and permutation of DC differential values; there are $\varphi_1 = h! \cdot 2^w \cdot \prod_{t=1}^{T}(|G^t|!)$ possible permutations in total. Here, $T$ is the number of groups of DC differential values and $|G^t|$ is the length of the group $G^t$. In the encryption of ACC phase, the ECSs excluding DCCs in inter-blocks are shuffled; thus, the possible permutations are $\varphi_2 = (h \cdot w)!$. To summarize, security can be ensured since it is very difficult to decrypt the encrypted JPEG bitstreams successfully by using only brute force, with a probability of $\varphi = \varphi_1 \cdot \varphi_2 = \frac{1}{h!\cdot 2^w \cdot \prod_{t=1}^{T}(|G^t|!)\cdot(h\cdot w)!}$. Table S1 presents the values of $\varphi_1$ and $\varphi_2$ for four 512×512 typical JPEG images with a QF of 80. As we can see, the key space of the proposed scheme, i.e., $\varphi$, is far above the secure key space of $2^{100}$ (Li et al., 2021), indicating that the proposed scheme can resist brute-force attacks.

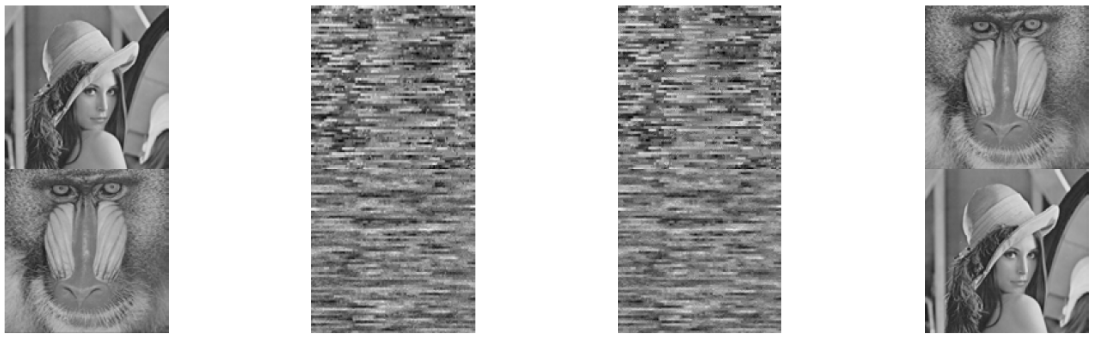
**Table S1 Key space for four typical JPEG images**

| JPEG image | $\varphi_1$ | $\varphi_2$ | $\varphi$ |
|---|---|---|---|
| Lena | $3.97\times10^{3057}$ | $3.64\times10^{13,019}$ | $1.15\times10^{16,077}$ |
| Couple | $4.13\times10^{3927}$ | $3.64\times10^{13,019}$ | $1.50\times10^{16,947}$ |
| Boat | $1.52\times10^{3861}$ | $3.64\times10^{13,019}$ | $5.55\times10^{16,880}$ |
| Baboon | $7.90\times10^{3753}$ | $3.64\times10^{13,019}$ | $2.88\times10^{16,773}$ |

Fig. S1 depicts the visual effect of JPEG images for Lena and Baboon at different stages. From the results, we can see that the valuable information of the original JPEG images is effectively masked after the processes of encryption and data embedding.

Table S2 demonstrates various statistical data of the marked encrypted JPEG images for Lena, Couple, Boat, and Baboon. From the results, we can see that the information entropy values of those four marked encrypted JPEG images are 7.7208, 7.7789, 7.7407, and 7.8230, respectively. The average information entropy is about 0.23 away from the maximum theoretical value 8.0, meaning that these marked encrypted JPEG images are somewhat random. The reason for such a difference is that a small portion of the correlation of DC coefficients after encryption has been maintained to achieve file size preservation; however, this results in a slight deviation in pixel distribution from complete randomness (Su et al., 2023). Hence, from a statistical perspective, the original JPEG image information is masked well. We also consider the NPCR (number of pixels change rate) and UACI (unified average changing intensity) (Su et al., 2023) values to measure the difference between

the original JPEG image and the marked encrypted JPEG image. As can be seen from Table S2 that the values of NPCR are extremely close to the maximum theoretical value of 1, meaning that the marked encrypted JPEG image is almost completely different from the original one. Moreover, the UACI value reaches 0.2743 on average and the PSNR is lower with an average value of 10.98 dB, which implies that the average changing intensity between the original JPEG image and the marked encrypted JPEG image is significant. Nevertheless, we can see that the UACI values are universally lower than a desired value of 0.3333, which means that there is room for future improvement of designing encryption methods. The reason for this is that, although DC differential values from a group are permuted, a small portion of the correlation of those DC coefficients has been maintained, especially in smooth regions. In such cases, for a given position, the difference in the DC coefficients before and after encryption exists but is not very large, resulting in a unified average changing intensity of decoded pixel values being sometimes finite. Additionally, Table S2 demonstrates a comparison of the correlation coefficients (Su et al., 2023) between the original and marked encrypted JPEG images. From the results, the correlation coefficient of each adjacent pixel pair in the original JPEG image is significantly higher than that in the marked encrypted JPEG image, which means that the correlation of the adjacent pixels is significantly broken.



(a) (b) (c) (d)

**Fig. S1 Visual effect for Lena and Baboon images when QF=85: (a) original JPEG image; (b) encrypted JPEG image; (c) marked encrypted JPEG image; (d) decoded JPEG image**

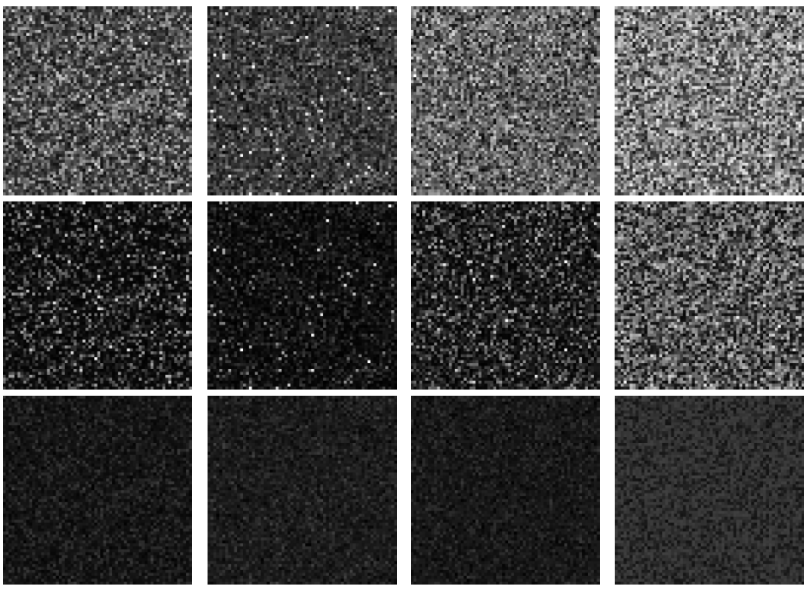
**Table S2 Statistical data for the four typical JPEG images**

| Image (QF=85) | Entropy | NPCR | UACI | PSNR (dB) | Horizontal | | Vertical | | Diagonal | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | $I_O$ | $I_M$ | $I_O$ | $I_M$ | $I_O$ | $I_M$ |
| Lena | 7.7208 | 0.9937 | 0.2817 | 10.76 | 0.9841 | 0.4441 | 0.9699 | 0.3288 | 0.9541 | 0.2787 |
| Couple | 7.7789 | 0.9942 | 0.2832 | 10.77 | 0.9538 | 0.2919 | 0.9395 | 0.3196 | 0.9031 | 0.1627 |
| Boat | 7.7407 | 0.9934 | 0.2810 | 10.39 | 0.9738 | 0.2213 | 0.9302 | 0.1867 | 0.9177 | 0.1359 |
| Baboon | 7.8230 | 0.9929 | 0.2511 | 12.00 | 0.7701 | -0.0251 | 0.8639 | 0.1159 | 0.7483 | 0.0612 |

Furthermore, we consider the sketch attack to catch the outline image of an original JPEG image directly from the corresponding marked encrypted version. In our experiments, we employ NCC (non-zero coefficient count), EAC (Energy of AC coefficients), and PLZ (Position of last non-zero AC coefficient) (Su et al., 2023) to evaluate the security of the resistance to sketch attacks. The outline images generated using NCC, EAC, and PLZ are illustrated in Fig. S2. As can be seen, no valuable outline information leaked out, which implies that the proposed scheme is safe against sketch attacks.

Finally, we make a comparison between the proposed scheme and the schemes of He JH et al. (2019), Hua et al. (2023), and Yuan et al. (2023), to clarify their differences in terms of security, as shown in Table S3. First, except for the initial key $K_{\text{int}}$, the knowledge of the iteration times in the schemes of He JH et al. (2019), Hua et al. (2023), and Yuan et al. (2023) should be shared between the encoder and decoder to ensure the perfect decryption. The visual content of the encrypted JPEG images becomes more blurred as the number of iterations increases. Also, more iterations will contribute to a larger key space. Second, the proposed scheme performs the region segment of DC coefficients in a larger region size, which may be conducive to thoroughly rearranging

DC coefficients when compared to the schemes (He JH et al., 2019; Hua et al., 2023; Yuan et al., 2023). This is one of the main reasons why the proposed scheme distorts the visual content well without the iterative grouping and swapping of DCCs. Finally, both the proposed scheme and the schemes of He JH et al. (2019), Hua et al. (2023), and Yuan et al. (2023) can defeat the chosen-plaintext attack using an adaptive encryption key generation strategy.



(a) (b) (c) (d)

**Fig. S2 Sketch attack analysis for the proposed scheme: (a) Lena; (b) Couple; (c) Boat; (d) Baboon. The top, middle, and bottom rows show the outline images generated using NCC, EAC, and PLZ, respectively**

**Table S3 Security comparison between the proposed scheme and other schemes**

| Item | He JH et al. (2019) | Yuan et al. (2023) | Hua et al. (2023) | Ours |
|---|---|---|---|---|
| Shared knowledge | $K_{\text{int}}$; Iteration number | $K_{\text{int}}$; Iteration number | $K_{\text{int}}$; Iteration number | $K_{\text{int}}$ |
| Key space | ** | ** | ** | * |
| Region size | * | * | * | ** |
| Adaptive encryption | Yes | Yes | Yes | Yes |

More * means larger key space or region size

## 2 Performance of file size preservation under various QFs

The performance of file size preservation under various QFs is illustrated in Fig. S3.

## 3 Comparison of PSNR among different schemes

Fig. S4 compares the PSNR of the marked approximate JPEG images provided by the proposed scheme with those of He JH et al. (2019)'s scheme. Among these, the marked approximate JPEG images are decrypted from the marked encrypted JPEG bitstreams directly, which naturally has some distortion. First of all, it is not surprising that the PSNR decreases as the payload increases, owing to the exchanges of RZL pairs in more DCT blocks. Second, the visual perception of the marked approximate JPEG images provided by both approaches is not ideal, especially in a relatively large payload. Finally, we can see from the results that, when the payload is

lower, our proposed scheme offers PSNR values of not less than 30 dB, which is considered to be reluctantly acceptable for human eyes.

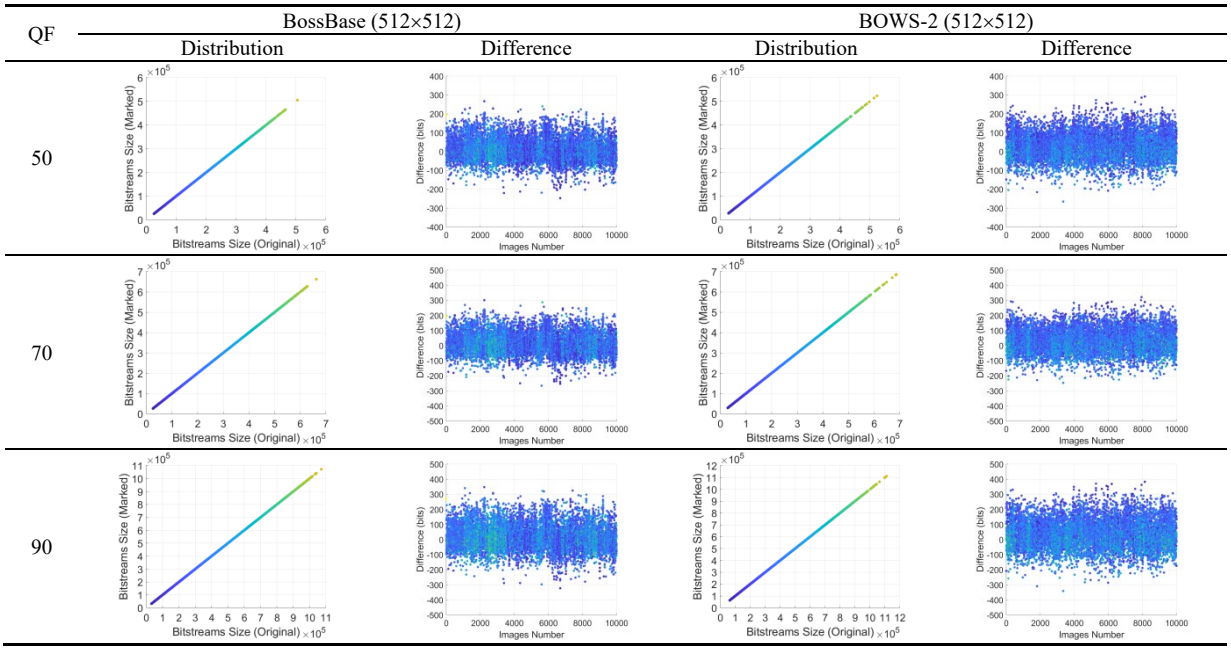



**Fig. S3 Illustration of the relationships of file size between the entropy encoded data of the original JPEG bitstreams and those of the marked encrypted JPEG bitstreams under various datasets and QFs**

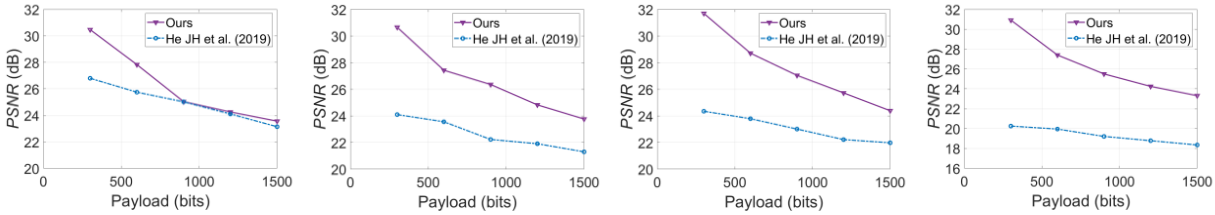



**Fig. S4 PSNR of the marked approximate JPEG images under various payloads when QF=85: (a) Lena; (b) Couple; (c) Boat; (d) baboon**

## 4 Comparison of payload and file size increment under various QFs

To further verify the embedding capacities of the related schemes with file size preservation under different QFs, we randomly select 200 images from the BossBase dataset and show their average payload and file size increment of the proposed scheme and Hua et al. (2023)'s scheme in Table S4. As we can see, in a sense, the proposed scheme is considered to have a good performance in terms of embedding capacity while keeping a lower increment in terms of file size.

## 5 Comparison of features for different RDH schemes in the encrypted compressed domain

To further demonstrate the usability of our proposed scheme, we compare the features of different RDH schemes in various encrypted compressed domains, as shown in Table S5. It is noted that the schemes of Yin

et al. (2018) and Cao et al. (2022) are RDH in encrypted AMBTC and VQ-encoded images, respectively. Similar to our proposed scheme, both schemes by Yin et al. (2018) and Cao et al. (2022) perform well in terms of features of file size preservation, format compatibility, and reversibility. For Puteaux et al. (2021)'s scheme, the DC overflow problem may exist because of their sign bit substitution based embedding mechanism. Simultaneously, the original sign bit is lost after embedding and then is expected to be reconstructed using a defined predictor with the help of correlations between neighboring blocks. Regrettably, it does not always work perfectly. With respect to embedding capacity, the performance of Yin et al. (2018)'s scheme goes on a downward in the complexity image because the number of the determined peak difference points exploited to embed data is reduced. In contrast, our proposed scheme and Cao et al. (2022)'s scheme (Codebook size is 128) have their merits and downsides, but have a higher value compared to that of the scheme of Puteaux et al. (2021).

**Table S4 Average payload and file size increment of our scheme and Hua et al. (2023)'s scheme on images from the BossBase dataset**

| QF | Ours | | Hua et al. (2023) | |
|---|---|---|---|---|
| | Payload | File size increment | Payload | File size increment |
| 50 | 3349.1 | 10.6 | 3041.2 | 184.3 |
| 60 | 3724.9 | 11.3 | 3932.9 | 234.7 |
| 70 | 4233.9 | 12.4 | 4667.2 | 251.2 |
| 80 | 5421.7 | 13.5 | 7536.1 | 292.4 |
| 90 | 6772.4 | 14.8 | 10937.2 | 332.5 |

**Table S5 Comparison of features for different RDH schemes in encrypted compressed domain**

| Feature | Yin et al. (2018) | Cao et al.( 2022) | Puteaux et al. (2021) | Ours |
|---|---|---|---|---|
| Domain | AMBTC | VQ | JPEG | JPEG |
| File size preservation | Yes | Yes | Yes | Yes |
| Format compatibility | Yes | Yes | No ideal | Yes |
| Reversibility | Yes | Yes | No always | Yes |
| Capacity (Lena) | 6336 bits | 4889 bits | 2010 bits | 5772 bits |
| Capacity (Baboon) | 1653 bits | 6835 bits | 2044 bits | 6357 bits |

## 6 Computational complexity analysis and comparison

Encryption phase: For the scheme by He JH et al. (2019), the computational cost of each encryption procedure is analyzed as follows: for group DCCs and permute consecutive DCCs with the same sign, the computational cost is $2 \cdot \Theta(L)$; for iterative group and swap DCCs, the computational cost is $\tau \cdot \Theta(L)$, where $\tau$ is the number of iterations; for permute ACCs with the same category, the computational cost is $\sum_{l=1}^{L} \Theta(K_l)$; for permute ECSs excluding DCCs, the computational cost is $\Theta(L)$. Due to the encryption process of the DC component in the schemes of Hua et al. (2023) and Yuan et al. (2023) being the same as that of He JH et al. (2019), their computational cost for the DC component is analyzed as $(2 + \tau) \cdot \Theta(L)$. As to Yuan et al. (2023)'s scheme, the AC component encryption includes the permutation of ACCs with the same frequencies and the same run length, and the ECSs' permutation excluding DCCs, so the computational cost can be analyzed as $\sum_{l=1}^{L} \Theta(K_l)\ /16 + \Theta(L)$. Hua et al. (2023)'s scheme encrypts ACAs using AES encryption and scrambles ACHs among blocks, and the computational cost can be roughly analyzed as $5 \cdot \sum_{l=1}^{L} \Theta(K_l) + \Theta(L)$. In the proposed scheme, the computational complexity mainly lies in the following aspects: for reorder and reverse DC coefficients, computational cost is $\Theta(L)$; for group and permute DC coefficients, computation cost is $2 \cdot \Theta(L)$; for recoding the DC coefficients, the computational cost is $\Theta(L)$; for permute inter-block ECSs excluding DCCs, the computational cost is $\Theta(L)$.

Data embedding phase: In the method of He JH et al. (2019) and the proposed scheme, the cost of determining embeddable blocks is $N \cdot \Theta(L)$, and the rotation-based data embedding algorithm is the most time-consuming operation, with a computational cost of $N \cdot \Theta(L)$. For Yuan et al. (2023)'s scheme, the computational cost of histogram shifting-based data embedding and frequencies and ECSs selection can be represented as $65 \cdot \Theta(L)$. In Hua et al. (2023)'s scheme, the cost of determining embeddable blocks is $2N \cdot \Theta(L)$, and their fast search implementation on the permutation operation of the multiset is the time-consuming operation and analyzed as $N^2 \cdot \Theta(L)$.

For parameter $\tau$, He et al. (2019)'s scheme states that the number of iterations is set to 15 in most cases; thus, $\tau$ is considered to be not less than 15, i.e., $\tau \geq 15$. Except that, $N \leq K_l \leq 63$ exists in general. Here, we assume that $K_l = 16$ (an experimental average value of 886 images from the UCID dataset when QF=85) and suppose that $N$ is infinitely close to $K_l$ to maximize the embedding capacity under the ideal condition. Based on this, the proposed scheme provides a lower computational cost than the schemes of He JH et al. (2019), Hua et al. (2023), and Yuan et al. (2023).

## References


Cao F, Fu Y, Yao H, et al., 2022. Separable reversible data hiding in encrypted VQ-encoded images. *Secur Commun Netw*, Article 1227926. https://doi.org/10.1155/2022/1227926

He JH, Chen JX, Luo WQ, et al., 2019. A novel high-capacity reversible data hiding scheme for encrypted JPEG bitstreams. *IEEE Trans Circ Syst Video Technol*, 29(12):3501-3515. https://doi.org/10.1109/TCSVT.2018.2882850

Hua ZY, Wang ZY, Zheng YF, et al., 2023. Enabling large-capacity reversible data hiding over encrypted JPEG bitstreams. *IEEE Trans Circ Syst Video Technol*, 33(3):1003-1018. https://doi.org/10.1109/TCSVT.2022.3208030

Li B, Feng Y, Xiong Z, et al., 2021. Research on AI security enhanced encryption algorithm of autonomous IoT systems. *Inform Sci*, 575: 379-398. https://doi.org/10.1016/j.ins.2021.06.016

Puteaux P, Wang ZC, Zhang XP, et al., 2021. Hierarchical high capacity data hiding in JPEG crypto-compressed images. Proc 28th European Signal Process Conf, p.725-729. https://doi.org/10.23919/Eusipco47968.2020.9287376

Su GD, Chang CC, Lin CC, et al., 2023. Towards property-preserving JPEG encryption with structured permutation and adaptive group differentiation. *Vis Comput*, 40:6421-6447. https://doi.org/10.1007/s00371-023-03174-5

Yin Z, Niu X, Zhang X, et al., 2018. Reversible data hiding in encrypted AMBTC images. *Multimed Tools Appl*, 77(14):18067-18083. https://doi.org/10.1007/s11042-017-4957-6

Yuan Y, He HJ, Chen F, et al., 2023. Reversible data hiding in encrypted JPEG image with changing the number of AC codes. *Multimed Tools Appl*, 82(28):43649-43669. https://doi.org/10.1007/s11042-023-14614-8