Cryptanalysis of a Self-Recovery Fragile Watermarking Scheme

Oussama Benrhouma† †††, Rhouma Rhouma††, and Ahmad Taleb†

†Faculty of Computer and Information Systems, Islamic University of Madinah, Madinah, Saudi Arabia
††College of Applied Science, Sallalah, Sultanate of Oman
†††Ecole Nationale d'Ingenieurs de Tunis (ENIT), Tunisia

Summary
In this paper, we analyze the security of a self-recovery fragile watermarking scheme proposed by C. Wang et al. An attack against C. Wang et al.'s scheme is demonstrated. The theoretical and experimental results show that the proposed scheme is not secure against attacks.

Keywords: *SVD, cryptanalysis, watermarking, tamper detection, attack.*

1. Introduction

Communications and technological inventions have witnessed a huge leap in the past two decades, the thing that allowed sophisticated devices such as digital cameras and smart-phones to emerge, and with the revolutions communications technologies, the access to public channel becomes easier and cheaper, add to that the emergence of social media and the rapid growth of its users around the globe. all of these factors made the world that we live in today, a digital world. Most people nowadays have easy access to the internet and the amount of data exchanged is huge. With this technological advancement emerge new challenges, and the challenge that we face today is how to secure our privacy and content integrity.

Most of the data exchanged over the internet today is a multimedia contents data, especially images, the need to secure these image become essential, Image is even presented today in courtrooms as evidence. The problem is that with the presence of so many easy to use software to manipulate images such as photoshop, the integrity of any image could be doubtful which raise the need for security schemes to ensure the integrity of the images.

Digital watermarking present a solution to control image integrity, many researchers have proposed watermarking schemes to detect and locate possible forgeries in the digital images [1],[5], however, the robustness of any security system against possible attacks should be tested to improve the quality and the security of these schemes.

In this paper a cryptanalysis of a watermarking scheme for tamper detection is presented, The rest of the paper is organized as follows: in section 2 a description of the scheme under study is presented [1], section 3 describes the steps leading to the cryptanalysis of the scheme and presents the results of the attack, and section 4 concludes the manuscript.

2. The scheme under study

2.1 The embedding process

The watermarking scheme proposed by Wang et al. in [1] could be briefly described as follows: Given an Image I with size M x M.

1. All 3 LSBs of the image are initialized to zeros.
2. The image is decomposed to non-overlapping blocks of size 2 x 2
3. For each block the singular value decomposition (SVD) is applied to produce a diagonal matrix S_i and two matrices orthogonal matrices U_i and V_i. Where i is the block number:
$$i = \frac{MxM}{2x2}$$
4. Each block is now processed to be judged as a smooth block or a texture block:
   (a) The SVD is applied for each block and the two orthogonal matrices U_i and V_i are extracted.
   $$B_i = U_i * S_i * V_i^T$$
   (b) A matrix is calculated R_i by the factorization of U_i and V_i,
   $$R_i = U_i * V_i^T$$
   (c) The resulted matrix is processed using two thresholds T_1 and T_2: for every pixel of the matrix R_i, if its value falls in the range of T_1 and T_2 the pixel will be judged as a smooth pixel, otherwise it is considered a texture pixel. T_1 and T_2 have fixed values: T_1 = 0.48 and T_2 = 0.52.
   (d) Finally the number of smooth pixels are calculated: if the number reaches 3 so the block Bi will be classified as smooth block, otherwise it will be considered as a texture block.

Manuscript received March 5, 2024
Manuscript revised March 20, 2024
https://doi.org/10.22937/IJCSNS.2024.24.3.5

After the classification of all blocks of the image into smooth and texture, the watermark is now generated. The watermark for each block includes two parts: an authentication bits and recovery bits.

5. For each block Bi the authentication watermark is generated using the singular matrix Si. The trace of the matrix is calculated using equation 1, then converted to binary form. $tr(Si) = b_1, b_2, ..., b_8, b_9$

$$tr(S_i) = \sum_{j=1}^{2} a_{i,j} = a_{1,1} + a_{2,2} \qquad (1)$$

To generate two bits authentication watermark p_1 and p_2, an exclusive-or-operations are applied as shown equation 6.

$$p_1 = b_1 \oplus b_2 \oplus ... \oplus b_8 \oplus b_9$$
$$p_2 = b_2 \oplus b_4 \oplus b_6 \oplus b_8 \qquad (2)$$

Finally a pseudo-random sequence N = (n1, n2) is generated using a key (K1) to encrypt the two bits authentication watermark.

$$W_a = (n_1 \oplus p_2,\ n_2 \oplus p_2)$$

6. A recovery watermark R is now generated for each block. The watermark has a variable capacity depending on the classification of the block (smooth or texture).

$$R= \begin{cases} 6\ bits \quad for\ smooth\ blocks \\ 10\ bits\ for\ texture\ blocks \end{cases}$$

(a) For the smooth blocks, the average value of the block is saved and coded into 5-bits to represent. The first bit of the watermark will take 0, representing the type of the block, and the rest 5-bits represent the average value of the block.

(b) For texture block, the first bit would be 1 to represent the type of the block, the rest 9 bits are generated as follows:

- The texture block Bi is preprocessed using equation 3

$$B_i' = \left\lfloor \frac{B_i}{2} - 8 \right\rfloor \qquad (3)$$

- The DCT transform is applied to the resulted block B_i' to obtain a DC coefficient and 3 AC coefficients.
- After rounding, the DC coefficient is encoded into 5 bits : 1 bit sign flag and 4 bits encoding result.
- The first Ac coefficient is encoded into 4 bits: 1 bit sign flag and 3 bits encoding result.

At this point, all 9 bits of the recovery watermark $R = r_2, ..r_9$ are calculated.

To obtain the final recovery watermark, a pseudo-random sequence N_2 is generated using a key K_2 and the watermark R is encrypted to obtain the encrypted recovery watermark Wr.

$$Wr = N_2 \oplus R \qquad (4)$$

7. The watermark is now ready to be embedded into the host image. The embedding is done as follows:
   a. The two bits authentication watermark Wa for each block are embedded into the two LSBs of the first pixel of the block itself.
   b. The recovery watermark Wr for each block are embedded into the 2 or 3 LSBs of its mapping block.
   The mapping function is defined by equation 5.

$$X' = (K \times X) modN + 1$$

(5)

Where:

- X and X' are the block index.
- N is the total number of blocks.
- K is a prime number where $K \in [1, N-1]$.

Note that the keys for this scheme are: K_1 and K_2 used in the generation of the pseudo-random sequence to encrypt the watermark, and K the key used in the mapping process. The flowchart of the embedding scheme is shown in figure 1.

2.2 The extraction process

The extraction and tamper detection process is done a 3 levels as follows:

1. The authentication watermark Wa is extracted then a pseudo-random sequence is generated using the secret key K_1 and the extracted watermark is decrypted to obtain the extracted decrypted watermark We.

   A new authentication watermark is regenerated in same way in step 5 in the embedding process to obtain a calculated watermark Wc. Finally, a comparison between the two watermarks We and Wc is done to determine if the block is tampered with or not.

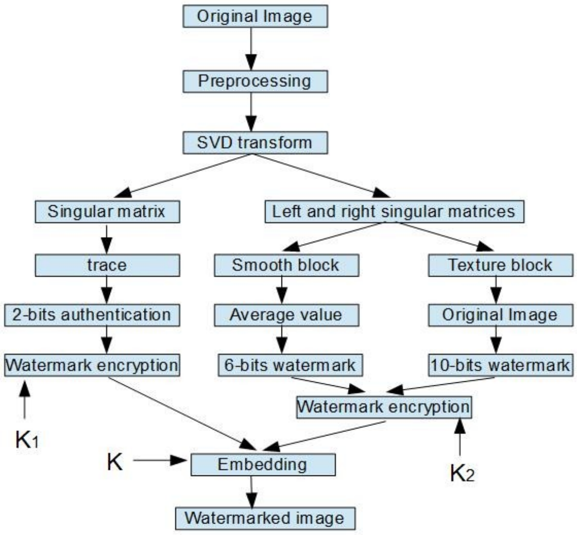

Fig.1 Diagram of the embedding process

2. The second level detection is done based on the recovery watermark as follows: After determining the type of the block (smooth or texture) the mapping function is used with the key K and the recovery bits of the block in question are extracted, then decrypted using the pseudo-random function with the key K_2. A new recovery bit is now calculated for the block and a comparison is conducted between the extracted watermark and the calculated one to determine if the block has been falsified.
3. The third level detection is based on the neighboring blocks : if a block is marked as tampered while less than 2 of his blocks neighbors are marked as tampered then the block is marked as valid, on the other hand if block is marked as valid while more then 7 of his neighboring blocks are marked as falsified so it will be marked as tampered.
   The flowchart of the extraction process is shown in figure 2.

2.2 The recovery process

After identification of tampered blocks, the recovery process is now executed: The mapping function is used to identify the mapping block of the tampered one, and the recovery bits are extracted and decrypted using the secret key K_2. and the pixels of the falsified block are replaced with the recovery value:

- If the block is a smooth block , 6-bits are extracted and used to recover the block average.
- if it is a texture block, 10-bits are extracted representing the DC and the first AC value, finally the inverse discrete cosine transform is applied to reconstruct the falsified block.

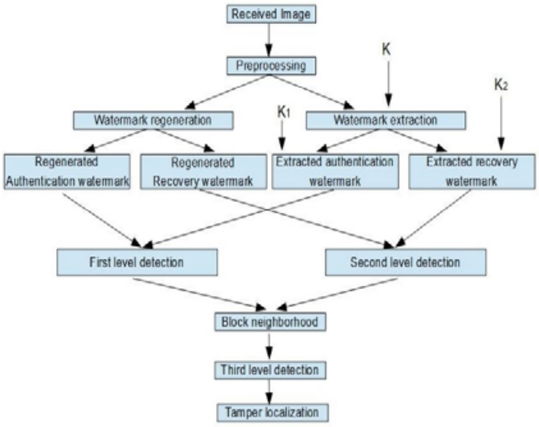

Fig.2 Diagram of the extraction process

3. Cryptanalysis of the scheme

To prove that the proposed scheme is not secure, a cryptanalysis on the proposed scheme [1] is conducted. The cryptanalysis is successful only if a falsification on watermarked image is done without being detected by the extraction scheme.

Based On kerchoff's principle [6], the security of a cryptosystem should be based only on the keys, everything else about the scheme should be known. In other words, a cryptanalyst knows everything about the cryptosystem except for the secret keys.

In Our case the keys are :
- The keys K1 and K2 used to generate a pseudo-random sequence to encrypt the watermarks.
- The key K used in the mapping function.

A cryptanalysis of the scheme in [1] is conducted in this paper without any previous knowledge of the secret keys. The propose a method to falsify an intercepted watermarked image without being detected by proposed scheme.

Based on kerckhoff's [6] we have temporary access to the watermarking machine, the first step in the attack is to conquer the mapping function and reveal the mapping position of each block, for that a chosen plaintext attack (CPA) is conducted:
To conquer the mapping function, we propose to use the chosen plain text attack technique (CPA):

1. An Image I is chosen with the same size of the intercepted one (M × M) where the values of all pixels is zero, except for one block (the targeted block), where its value is set randomly (but not zero of course).
2. The image I is the injected to the watermarking machinery to get the watermarked image $_{Iw}$.
3. The block of the resulted watermarked image $_{Iw}$ should have the same value except for two blocks: the chosen one and its mapping position.

4. These steps are repeated $\frac{B_n \ x \ B_n}{2}$ times to reveal all the mapping positions of the image blocks, where $_{\mathrm{Bn}}$ is the number of blocks:

$$B_n = \frac{MxM}{2x2}$$

Numerical example of the mapping block technique is shown in the following equation.

$$\left(\begin{array}{cc|cc} 222 & 222 & 0 & 0 \\ 222 & 222 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right) \Rightarrow \left(\begin{array}{cc|cc} 220 & 223 & 3 & 3 \\ 222 & 221 & 3 & 3 \\ \hline 3 & 3 & 2 & 7 \\ 3 & 3 & 6 & 3 \end{array}\right)$$

5. After revealing all blocks index positions, the secret pseudo-random sequences N and N_2 could now be calculated. Note that N and N_2 are used to encrypt the authentication watermark and the recovery watermark respectively.
The keys are calculated as follows:
   (a) Using known plain-text attack, an image is injected to the watermarking machinery to obtain the watermarked image W
   (b) A classification of blocks to smooth and texture is done in the same way as in the embedding scheme, then the authentication watermark $w_a = (a_1, a_2)$ and the recovery watermark Wr are extracted.
   (c) for each block the steps 1 - 5 of the embedding process are applied to obtain two bits authentication watermark p_1 and p_2.
   (d) The first pseudo-random sequence $N = (n_1, n_2)$ could now be calculated using equation 7.

$$N = \begin{aligned} n_1 &= a_1 \oplus p_1 \\ n_2 &= a_2 \oplus p_2 \end{aligned} \qquad (7)$$

   (e) Based on the classification of the block, the recovery watermark W_r for the target block is extracted from its mapping position, then step 6 of the embedding scheme is applied to obtain the recovery bits R.
   (f) The pseudo-random bits N_2 could now be calculated using equation 8.

$$N_2 = R \oplus W_r \qquad (8)$$

6. A falsification could be done now on the intercepted watermarked image:
   - The image is falsified with any type of attack.
   - The 3-LSBs of the falsified image are initiated to zeros.
   - Image is divided to blocks of size 2 × 2.
   - For every block a 2-bit authentication watermark is calculated and encrypted using the calculated pseudo-random sequence *N*.
   - The corresponding recovery bits for each block are then calculated and encrypted using the calculated encryption key N_2.

The results of the attack are shown in figures 3, 4 and 5.

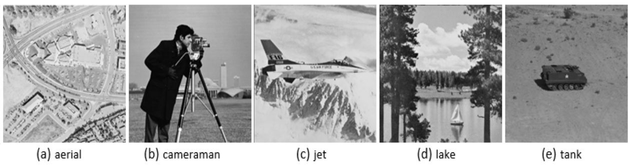
(a) aerial (b) cameraman (c) jet (d) lake (e) tank

Fig. 3 Original images

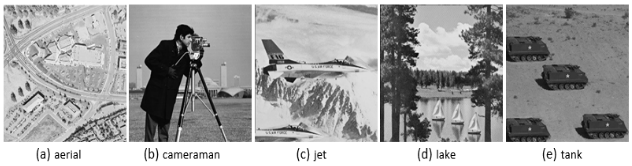
(a) aerial (b) cameraman (c) jet (d) lake (e) tank

Fig. 4 Tampered images

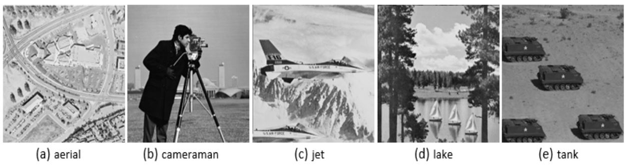
(a) aerial (b) cameraman (c) jet (d) lake (e) tank

Fig. 5 Tamper images

4. Conclusion

In this paper a successful cryptanalysis of a watermarking scheme is conducted, the watermarked images undergo falsifications without being detected by the extraction technique. This work is conducted to prove that the security of a scheme don't rely only on the randomness of the keys but in the design and the way these pseudo-random functions are used.

References

[1] Chengyou Wang, Heng Zhang, and Xiao Zhou. A self-recovery fragile image watermarking with variable watermark capacity. Applied Sciences, 8(4), 2018.

[2] Oussama Benrhouma, Houcemeddine Hermassi, and Safya Belghith. Tamper detection and self-recovery scheme by dwt watermarking. Nonlinear Dynamics, 79(3):1817 – 1833, Feb 2015.

[3] Javier Molina-Garcia, Beatriz P. Garcia-Salgado, Volodymyr Ponomaryov, Rogelio Reyes-Reyes, Sergiy Sadovnychiy, and Clara Cruz-Ramos. An effective fragile watermarking

scheme for color image tampering detection and self-recovery. Signal Processing: Image Communication, 81:115725, 2020.

[4] Behrouz Bolourian Haghighi, Amir Hossein Taherinia, and Amir Hossein Mohajerzadeh. Trlg: Fragile blind quad watermarking for image tamper detection and recovery by providing compact digests with optimized quality using lwt and ga. Information Sciences, 486:204 – 230, 2019.

[5] Durgesh Singh and Sanjay K. Singh. Effective self-embedding watermarking scheme for image tampered detec- tion and localization with recovery capability. Journal of Visual Communication and Image Representation, 38:775 – 789, 2016.

[6] A Kerckhoffs. La cryptographie militaire. Journal des sciences militaires, 9:5–38, 1883.