



ELSEVIER

Pattern Recognition Letters 23 (2002) 1847–1852

Pattern Recognition
Letters

www.elsevier.com/locate/patrec

Cryptanalysis of an encryption scheme for binary images

Chin-Chen Chang ^{*}, Tai-Xing Yu

*Department of Computer Science and Information Engineering, National Chung Cheng University,
Minghsiang, 621 Chiayi, Taiwan ROC*

Received 21 March 2001; received in revised form 8 February 2002

Abstract

In 1998, Chung and Chang proposed a new method to encrypt binary images. When compared to its predecessors, their method can actually obtain better compression ratio by putting different scan patterns at the same level in the scan tree structure and using the two-dimensional run-encoding technique. However, their scheme seems to be quite vulnerable to attacks if it uses the same key to encrypt different images. In this paper, we show that their scheme can be broken with some pairs of plain image and cipher image.

© 2002 Elsevier Science B.V. All rights reserved.

Keywords: Image encryption; SCAN language; 2DRE technique; Cryptography; Cryptanalysis

1. Introduction

In this modern, hi-tech world, network communication becomes more and more popular. Because of this there is a greater need for information security than ever before. In order to transmit secret images to other people, a variety of encryption schemes have been proposed. Bourbakis et al. (Bourbakis, 1986, 1997; Bourbakis and Alexopoulos, 1992, 1993) presented some image encryption methods using SCAN language. Recently, Chung

and Chang (1998) presented a new method for binary image encryption using modified SCAN language. By putting different scan patterns at the same level in the scan tree structure and employing a two-dimensional run-encoding (2DRE) technique, they pointed out that their scheme can encrypt images with higher security which results in a better compression ratio when compared to its predecessors (Bourbakis and Alexopoulos, 1992; Chang and Liu, 1994).

For a good encryption scheme, it is reasonable that the encryption key would be used for a period of time, otherwise the scheme will spend a lot of overheads on transferring the encryption key. Based on this assumption, we will show that Chung and Chang's scheme is vulnerable to attacks when the same key is used to encrypt different images. We use the known-plaintext attack method similar to

^{*} Corresponding author. Tel.: +886-5-272-0411; fax: +886-5-272-0859.

E-mail addresses: ccc@cs.ccu.edu.tw (C.-C. Chang), yts88@cs.ccu.edu.tw (T.-X. Yu).

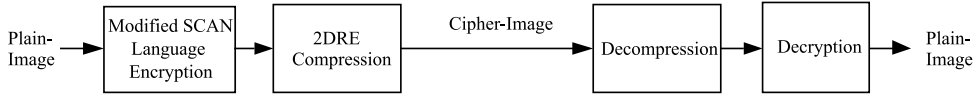


Fig. 1. Block diagram of the encryption and decryption scheme.

the method proposed in (Jan and Tseng, 1996). That is, after acquiring some plain-image and cipher-image pairs, we can break their scheme without generating all the combinations of the scan patterns. In this paper, we also present an example to illustrate our scheme of attack.

The remainder of this paper is organized as follows. In Section 2, we shall give a brief review of the image encryption scheme proposed by Chung and Chang. In Section 3, we shall present the cryptanalysis of Chung and Chang's encryption method. An example of the attack process will be also illustrated in this section. In Section 4, the conclusions will be given.

2. Review of the image encryption scheme

In the following, we review the image encryption scheme proposed by Chung and Chang (1998). The block diagram of Chung and Chang's encryption scheme is shown in Fig. 1. We will go through Chung and Chang's modified SCAN language which allows different scan patterns to be put at the same level in the scan quadtree structure. We suggest that interested readers consult (Chung and Chang, 1998) for more details of quadtree structure. After reviewing the modified SCAN language, we will give a simple example to illustrate the encryption process. Since the 2DRE technique has nothing to do with the security analysis, we do not discuss it in this section.

2.1. The modified SCAN language

The modified SCAN language for a $2^n \times 2^n$ binary image can be described by the 4-tuple (V_N, V_T, P, S) , where S is the starting symbol, P is the set of production rules, $V_N (= \{S, \bigcup_{i=1}^n L_i\})$

is the set of nonterminal symbols, and $V_T (= \{\bigcup_{i=1}^n R_j^i | 1 \leq j \leq 4^{i-1}\})$ is the set of terminal symbols (L_i denotes the set of different scan patterns at the i th level in the quadtree and R_j^i is one of the twenty-four scan patterns defined in Fig. 2). Each pattern SP_k , where $0 \leq k \leq 23$, is the k th scanning method for the 2×2 pattern window (Chung and Chang, 1998).

Based on the definition of the modified SCAN language, we can encrypt the binary image in Fig. 3 by applying the following production rule to its corresponding quadtree in Fig. 4:

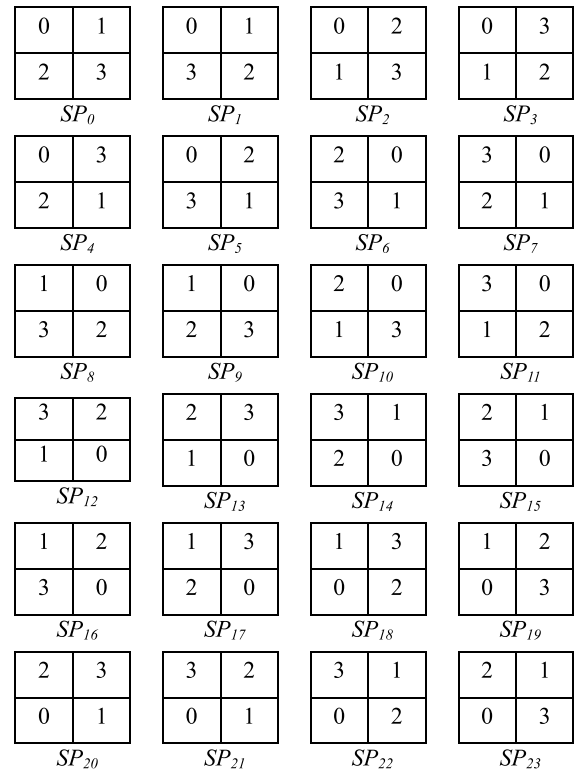
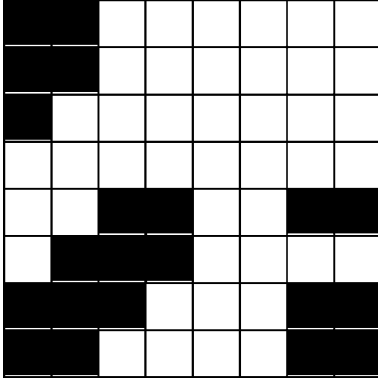


Fig. 2. The twenty-four scan patterns.

Fig. 3. A $2^3 \times 2^3$ binary image.

$$S \rightarrow L_1 L_2 L_3$$

$$L_1 \rightarrow R_1^1$$

$$L_2 \rightarrow R_1^2 R_2^2 R_3^2 R_4^2$$

$$L_3 \rightarrow R_1^3 R_2^3 R_3^3 R_4^3 \cdots R_{16}^3$$

$$R_1^1 = SP_1$$

$$R_1^2 = SP_3, \quad R_2^2 = SP_1, \quad R_3^2 = SP_{15}, \quad R_4^2 = SP_{14}$$

$$R_1^3 = SP_0, \quad R_2^3 = SP_4, \quad R_3^3 = SP_{12}, \quad R_4^3 = SP_9$$

$$R_5^3 = SP_1, \quad R_6^3 = SP_0, \quad R_7^3 = SP_{10}, \quad R_8^3 = SP_8,$$

$$R_9^3 = SP_{14}, \quad R_{10}^3 = SP_2, \quad R_{11}^3 = SP_{15}, \quad R_{12}^3 = SP_7$$

$$R_{13}^3 = SP_{21}, \quad R_{14}^3 = SP_3, \quad R_{15}^3 = SP_5, \quad R_{16}^3 = SP_6$$

After applying the production rule to the quadtree in Fig. 4, the resultant encrypted quadtree is shown in Fig. 5. Using the encrypted quadtree, the encrypted image in Fig. 6 is displayed by using the raster scanning method.

Up to now, we have shown the encryption method proposed by Chung and Chang. After

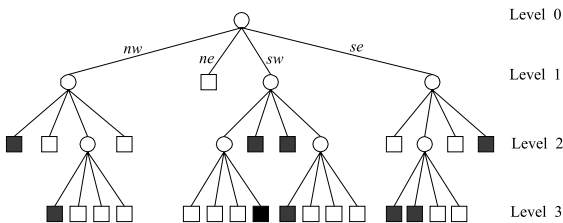


Fig. 4. The corresponding quadtree of Fig. 3.

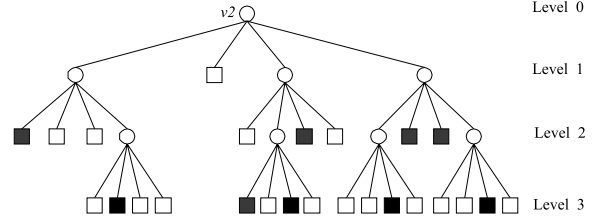


Fig. 5. The encrypted quadtree.

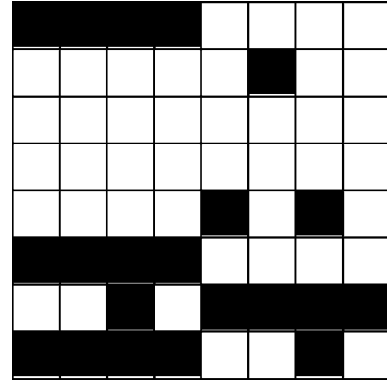


Fig. 6. The encrypted image by using the raster scanning method.

encryption, the cipher image is constructed by compressing the encrypted image.

To analyze the security of this encryption method, Chung and Chang mentioned that the security of an encrypted image depends on the total number of combinations employed in the encrypted image. Using their method, we find that the total number of possible scan pattern combinations for a $2^n \times 2^n$ encrypted image is $N(n) = 1 \times 24 \times 24^4 \times \cdots \times 24^{4^n-1} = 24^{(4^n-1)/3}$. In other words, the security of a $2^n \times 2^n$ encrypted image is equal to $N(n)$ and the computational complexity for an opponent to break this encrypted image is $O \times (N(n)) = O(24^{4^n})$. Here, we assume that one computation can perform one combination for putting one specific scan pattern. Hence the time needed to generate $N(n)$ scan patterns for a computer with the computational capability of 100 MIPS (million instructions per second) is $T(n) = N(n)/100 \times 10^6 \times 60 \times 60 \times 24 \times 365$ (years), which also shows the time needed to break a $2^n \times 2^n$ encrypted image.

3. Cryptanalysis of image encryption scheme

In this section, we present our scheme of attacking the image encryption scheme. Suppose we get a $2^n \times 2^n$ plain-image P_{nn} and its corresponding cipher-image C , which is compressed by a 2DRE technique. We can then decompress C and find the possible encryption rules. In order to decompress C , whose size is m bits, we have to decide how many bits are needed to represent each length in a binary representation. Assume that C has the highest compression rate and each length is represented by a fixed-length binary representation with l bits. We can determine l by the following procedure:

- Find the first non-zero bit in C . Here we suppose it is the k th ($k \leq 2l$) bit.
- Compute $L = \{l' | l' \text{ is a divisor of } m, 1 \leq l' \leq k\}$.
- For each element l' in L , let the number of segmented pixel-strings be $s' = m/l' - 1$. Decompress C with the information s' and l' , we get a sequence of binary strings. Let r_i ($1 \leq i \leq s'$) denote the decimal representation of the $(i+1)$ th binary string. Let $r_{\max}$ be the maximum of r_i 's. If l' does not satisfy all of the following constraints, remove l' from L .

1. $l' = \lfloor \log_2 r_{\max} \rfloor + 1$.
2. $r_1 > 0, r_2 > 0, r_3 > 0, \dots, r_{s'} > 0$.
3. The number of black pixels in the decompressed image is equal to that in the plain-image P_{nn} .
4. The number of white pixels in the decompressed image is equal to that in the plain-image P_{nn} .

For example, using the 2DRE technique, the compressed representation of Fig. 6 is $\{1, 4, 9, 1, 22, 1, 1, 1, 1, 4, 6, 1, 1, 8, 2, 1, 1\}$. In this representation, 22 is the maximum length among all of the lengths of the segmented pixel strings. Again in the compressed representation, each element can be represented by $l = 5$ bits. Therefore, the size of the compressed cipher image C is $m = 5 \times 17 = 85$ bits. After receiving C , we can use the above pro-

cedure to determine l . First, we find that the first non-zero bit in C is the fifth bit, so $k = 5$ and $L = \{1, 5\}$. Consider the first case $l' = 1$, we get $s' = 85/1 - 1 = 84$. After decompressing C with the information $l' = 1$ and $s' = 84$, we can easily verify that not all of $r_i > 0$, $1 \leq i \leq 84$. Thus, we conclude that $l \neq 1$. In the second case, $l' = 5$, we find that $s' = 16$ and the decompressed image satisfies all of the constraints. So $l = 5$ and we get the decompressed cipher image.

In our experiments, we find that the above procedure can uniquely determine the correct l with very high probability. But in some special cases, we cannot uniquely determine l . That is, some incorrect l 's also satisfy all of the constraints in the above procedure. However, the probability that these cases will occur is smaller than $10^{-\alpha}$, $\alpha > 4$ for $n \geq 2$. To obtain the probability, we construct a program to test possible patterns of $2^n \times 2^n$ different blocks. For $n = 2$, there are $2^{2 \times 2^2} = 2^{16}$ possible patterns. We test all of these patterns and find that there are only four patterns having two possible l . Thus the probability of the patterns having incorrect l is $4/2^{16}$, which is smaller than $10^{-\alpha}$, $\alpha > 4$. For $n > 2$, since the number of possible patterns is very large, we cannot test all possible patterns. Therefore, we randomly test $2^{16} \times 2^n$ patterns for $2 < n < 9$ a lot of times. In our experiments, even though we could not exactly determine the number of these special patterns because they are not regular, we find that the number of these special patterns is very sparse and far smaller than 2^{2n} . Hence the probability of the patterns having incorrect l is smaller than $2^{2n}/2^{16} \times 2^{2n} = 2^{-16}$. According to these results, we conjecture that the probability of the patterns having incorrect l is smaller than $10^{-\alpha}$, $\alpha > 4$. Additionally, since the number of l 's in these cases is only a constant, we have the ability to determine the correct l in the following procedure without increasing too much computational complexity.

After decompressing C , let the decompressed cipher image be E_{nn} . We can find the possible encryption rule W following the procedure below:

- Construct the corresponding quadtree Q_p of the plain-image P_{nn} .

- Use E_{nn} to reconstruct the encrypted quadtree Q_e by applying the reverse processes (since E_{nn} is an image constructed from the encrypted quadtree Q_e using the raster scanning method).
- Compute the sum of all the pixel values for each node in Q_p and Q_e .
- Compute the permutation rule $R_j^i (1 \leq i \leq n, \leq j \leq 4^{i-1})$ by comparing the relationship between the sum lists for block j at level i in Q_p and Q_e . If there does not exist any permutation rule such that the sum lists for block j at level i in Q_p is equal to that in Q_e we realize that this E_{nn} is constructed from an incorrect l . So we have to restart the procedure with another E_{nn} .
- After one R_j^i is determined, we transform Q_e by employing the permutation rule R_j^i . By collect-

ing these permutation rules we can get the possible encryption rule W .

For example, we use the $2^3 \times 2^3$ binary image in Fig. 3 as the plain-image P_{33} and the image in Fig. 6 as the encrypted image E_{33} . From these two images, we can construct their corresponding quadrees Q_p and Q_e with the sum of their pixel values, as shown in Fig. 7. For Level 1 of the tree in Fig. 7(a) and (b), we find that the sums listed in Q_p and Q_e are $\{5, 0, 10, 6\}$ and $\{5, 0, 6, 10\}$, respectively. Using this information, we realize that the scan pattern for Level 1 is SP_1 and that $R_1^1 = SP_1$. After R_1^1 is determined, the new Q_e is shown in Fig. 7(c). At Level 2, we compute R_1^2, R_2^2, R_3^2 and R_4^2 , respectively. Using $\{4, 0, 1, 0\}$ and $\{4, 0, 0, 1\}$,

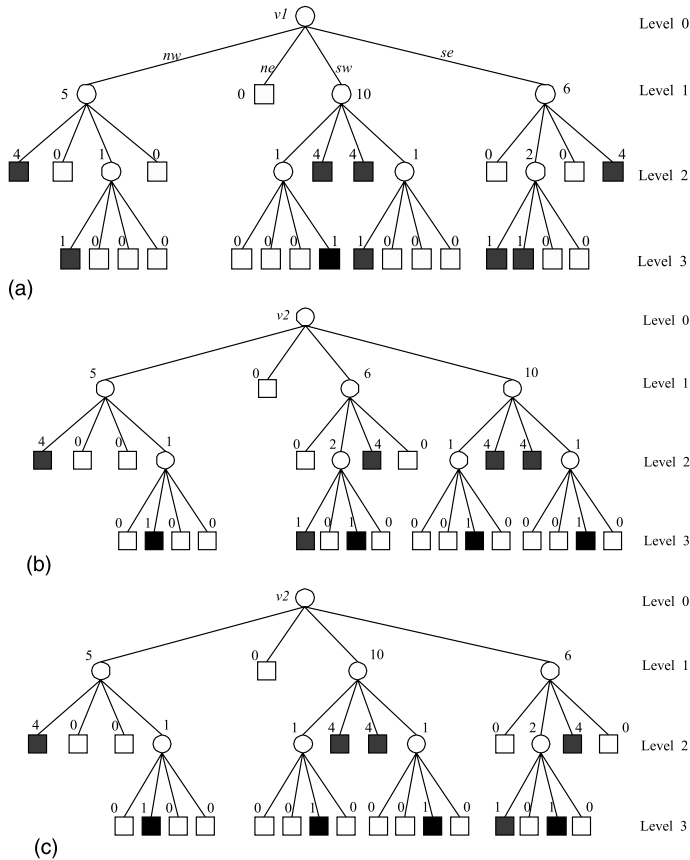


Fig. 7. The quadtree structure: (a) the quadtree Q_p with sum of pixel values in it, (b) the quadtree Q_e with sum of pixel values in it and (c) the quadtree Q_e after determining R_1^1 .

we find that $R_1^2 = SP_1$ or SP_3 . Since there is not enough information to determine R_1^2 uniquely, we denote $R_1^2 = SP_1 | SP_3$, where “|” denotes “OR” operation. Then we repeat the above steps recursively until all of the R_j^i 's are found. By collecting these R_j^i 's, we get the possible encryption rule W .

Now let us analyze the computational complexity of our scheme of attack. First, the time needed to decompress C is $O(m) \leq O(4^n)$. After constructing the quadtrees Q_p and Q_e , we have to determine the permutation rules R_j^i 's, where the total number of these R_j^i 's for a $2^n \times 2^n$ image is $R(n) = 1 + 4 + 4^2 + 4^3 + \dots + 4^{n-1} = (4^n - 1)/3$. For each R_j^i , we perform 24 comparisons between the sum lists for block j at level i in Q_p and Q_e , and constant time on the transforming Q_e . Hence the total computational complexity for determining one possible encryption rule W under our scheme of attack is $O(4^n)$.

It is easy to observe that we can efficiently decrease the number of possible encryption rules. However, sometimes we do not have enough information to uniquely determine one permutation rule R_j^i , so we need another pair of plain image and cipher image to help us to determine it. As the previous example illustrated, we only know that the permutation rule R_1^2 is SP_1 or SP_3 and cannot determine the original encryption rule. We need another pair of plain image P' and cipher image C' to help us. Assume that the permutation rule R_1^2 in P' and C' is $SP_2 | SP_3$. We can find that R_1^2 is SP_3 by intersecting these results. If we can get enough pairs of plain image and cipher image to determine each R_j^i , we can uniquely obtain the encryption rule. In the worst case, this happens when one pair of plain image and cipher image can only determine one permutation rule. The number of pairs we need is the number of undetermined permutation rules. However, if the cipher image is a chessboard-like image, it is hard to find the encryption rule with the above method. But this case does not occur usually. Therefore, we have shown that the encryption scheme proposed by Chung and Chang is vulnerable to attacks when they use the same key to encrypt different images.

4. Conclusions

In this paper, we have proposed a scheme of attack upon the image encryption system proposed by Chung and Chang (1998). By acquiring enough number of pairs of plain image and cipher image, we can use the known-plaintext attack to find the encryption rule. Thus, Chung and Chang's image encryption scheme is vulnerable to the known-plaintext attack. To avoid the known-plaintext attack, we offer some suggestions below. First, our scheme of attack can be proceeded successfully because the encryption rule shows the locality property. Therefore, we suggest that the encryption scheme should eliminate the locality property. Here locality property means that the number of black pixels in a block remains the same after encryption. In other words, the encryption scheme should make the cipher image as chaotic as possible. Second, we suggest that the order of encryption and compression stages in the scheme should be altered. By changing the order of these processes, the encryption scheme can achieve higher entropy and increase the difficulty of known-plaintext attack against this system.

References

- Bourbakis, N., 1986. A language for effective accessing of a 2D array. Proceedings of IEEE Workshop on LFA, Singapore, pp. 52–58.
- Bourbakis, N., 1997. Image data encryption and compression using G-SCAN. IEEE Conference on System, Man and Cybernetics, Orlando, FL, vol. 2, pp. 1117–1120.
- Bourbakis, N., Alexopoulos, C., 1992. Picture data encryption using scan patterns. Pattern Recognition 25 (6), 567–581.
- Bourbakis, N., Alexopoulos, C. 1993. Image data compression using SCAN patterns. Proceedings of SPIE—The International Society for Optical Engineering, vol. 1903, pp. 255–265.
- Chang, K.C., Liu, J.L., 1994. An image encryption scheme based on quadtree compression scheme. Proceedings of International Computer Symposium, Taiwan, pp. 230–237.
- Chung, K.L., Chang, L.C., 1998. Large encrypting binary images with higher security. Pattern Recognition Letters 19, 461–468.
- Jan, J.K., Tseng, Y.M., 1996. On the security of image encryption method. Information Processing Letters 60, 261–265.