Journal of Zhejiang University SCIENCE
ISSN 1009-3095
http://www.zju.edu.cn/jzus
E-mail: jzus@zju.edu.cn

Decryption of pure-position permutation algorithms*

ZHAO Xiao-yu (赵晓宇)[†1], CHEN Gang (陈 刚)[1], ZHANG Dan (张 亶)[3],
WANG Xiao-hong (王肖虹)[1], DONG Guang-chang (董光昌)[2]

([1]*Institute of DSP and Software Techniques, Ningbo University, Ningbo 315211, China*)
([2]*Institute of Imaging and Computer Graphics, Zhejiang University, Hangzhou 310027, China*)
([3]*Computer Science College, Zhejiang University, Hangzhou 310027, China*)
[†]E-mail: zhxiaoyu@math.zju.edu.cn

Received Feb. 12, 2004; revision accepted Mar. 24, 2004

Abstract: Pure position permutation image encryption algorithms, commonly used as image encryption investigated in this work are unfortunately frail under known-text attack. In view of the weakness of pure position permutation algorithm, we put forward an effective decryption algorithm for all pure-position permutation algorithms. First, a summary of the pure position permutation image encryption algorithms is given by introducing the concept of ergodic matrices. Then, by using probability theory and algebraic principles, the decryption probability of pure-position permutation algorithms is verified theoretically; and then, by defining the operation system of fuzzy ergodic matrices, we improve a specific decryption algorithm. Finally, some simulation results are shown.

Key words: Decryption, Image encryption, Fuzzy, Position permutation
Document code: A **CLC number:** TP391

INTRODUCTION

Due to the development of communication technology, information security becomes an increasingly important problem. The wide use of multimedia technology and the improvement in network transmission gradually enable direct acquisition of information clearly through images. Hence, data security has become a critically important issue. Many encryption algorithms have been proposed to protect valuable data from unauthorized parties.

The basic methods can be classified into three major categories: position permutation (Bourbakis and Alexopoulos, 1992; Matlas and Shamir, 1992; Ding and Qi, 1998), value transformation (Kuo and Chen, 1991), and the combining form (Yen and Guo, 1999; Sridharan *et al*., 1991; Yang and Kim, 1996; Lin and Klara, 1998; Refregier and Javidi, 1995). The position permutation, e.g. Zig-Zag, Arnold, Magic square transformation and other algorithms, merely moves the positions of the pixels in the original image in order to get the effect of encryption. Zhao and Chen (2002) introduced the concept of ergodic matrix, and used it to uniformly present scramble algorithms based on pixel shifting. However, there are still some potential weak points existing in these pure-position permutation algorithms, which are frail under known-text attack.

Obviously, the principle of transforming pixel positions is easy to find when just to compare the pixels' position in the image before and after en-

* Project supported by the National Natural Science Foundation of China (Nos. 60302012, 60202002) and the Youth Scientific Research Foundation of Ningbo (No. 2003A61006), China

cryption. However, because different pixels' positions of the image may have the same gray value, the larger the probability of the repetition, the more difficult the decryption will be. To deal with the weakness of pure position permutation algorithms, Lin and Klara (1998) proposed an improved decryption algorithm, which recovers the corresponding algorithm by comparing the original image and encrypted image.

PURE-POSITION PERMUTATION ENCRYPTION ALGORITHMS

Pure-position permutation algorithms, which are simple and rapid, are widely used in many image encryption systems. By introducing the concept of ergodic matrices, Zhao and Chen (2002) unified all the pure position permutations in a uniform frame and relevant combing operation to help realize their relevant permutation in an easy way, which means all the pure-position permutation algorithms can be realized through ergodic matrices.

Consider the gray scale images first, and denote the original image as $\boldsymbol{I}$, the encrypted image as $\boldsymbol{E}$, then the following formula is obtained:

$$\boldsymbol{I}=\begin{bmatrix} I_{11} & I_{12} & \cdots & I_{1n} \\ I_{21} & I_{22} & \cdots & I_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ I_{m1} & I_{m2} & \cdots & I_{mn} \end{bmatrix},$$

$$\boldsymbol{E}=\begin{bmatrix} E_{11} & E_{12} & \cdots & E_{1n} \\ E_{21} & E_{22} & \cdots & E_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ E_{m1} & E_{m2} & \cdots & E_{mn} \end{bmatrix}.$$

Unify all the pure position permutations in a uniform frame

(1) Ergodic matrix

An $m\times n$ matrix $\boldsymbol{R}$ is defined as an ergodic matrix, if every element of which is in the set of $\{1, 2, \ldots, mn\}$ and $r(i,j)=r(i',j')$, if and only if $i=i'$, $j=j'$. Note that $r_{(i-1)n+j}=r(i,j)$.

The following matrix is defined as a main ergodic matrix.

$$\begin{pmatrix} 1 & 2 & 3 & \cdots & n \\ n+1 & n+2 & n+3 & \cdots & 2n \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ (m-1)n+1 & (m-1)n+2 & (m-1)n+3 & \cdots & mn \end{pmatrix}_{m\times n}$$

(2) Realizing permutation by ergodic matrix

A rapid scrambling of image pixel permutation can be realized by ergodic matrix.

For instance, we can make an Ergodicity of the digital image matrix through the ergodic pattern represented by $\boldsymbol{R}_{m\times n}$ and arrange the results line by line. Hence, an image data permutation is realized.

Example:

$$\boldsymbol{I}_{4\times 4}=\begin{bmatrix} I_{11} & I_{12} & I_{13} & I_{14} \\ I_{21} & I_{22} & I_{23} & I_{24} \\ I_{31} & I_{32} & I_{33} & I_{34} \\ I_{41} & I_{42} & I_{43} & I_{44} \end{bmatrix}_{4\times 4} \xleftrightarrow{\boldsymbol{R}=\begin{bmatrix} 7 & 2 & 10 & 3 \\ 15 & 1 & 9 & 11 \\ 4 & 6 & 8 & 13 \\ 14 & 12 & 16 & 5 \end{bmatrix}} \begin{bmatrix} I_{22} & I_{12} & I_{14} & I_{31} \\ I_{44} & I_{32} & I_{11} & I_{33} \\ I_{23} & I_{13} & I_{24} & I_{42} \\ I_{34} & I_{41} & I_{21} & I_{43} \end{bmatrix}_{4\times 4}$$

Weakness of pure-position permutation

Any scramble algorithm can be expressed by ergodic matrix (Zhao and Chen, 2002). What is most important in decrypting a position permutation algorithm is to find its corresponding ergodic matrix.

The basic springboard of the algorithm:

Obviously, it is impossible to recover the corresponding ergodic matrix of a scramble algorithm with just one encryption image pair (Namely, the original image and the corresponding encrypted image using a certain pure-position permutation encryption algorithm). This is because the information in one encryption image pair is not enough for recovering the ergodic matrix. In contrast, with a series of encryption image pairs, it will be able to possible the ergodic matrix $\boldsymbol{R}$ gradually by synthesizing the information contained in all the encryption image pairs.

Problems to be resolved

In order to theoretically verify the feasibility of this approach, we must consider the following issues:

1) How many encryption image pairs are needed to make the recovered $\boldsymbol{R}$ clear enough to decrypt the encrypted images? Section 3 answered this question by simulation results.

2) How to express the form of the fuzzy $\boldsymbol{R}$, since all the factors of position transform are uncertain? In Section 3, we will fulfill this task by introducing the concept of fuzzy ergodic matrix.

3) How to accumulate the information of the position contained in each image pair? The accumulation is accomplished in Section 3 by defining the intersection of $\tilde{\boldsymbol{R}}$, and using several clearance methods for transforming $\tilde{\boldsymbol{R}}$ into $\boldsymbol{R}$.

OPERATION SYSTEM OF FUZZY ERGODIC MATRICES

First, the concept of fuzzy ergodic matrix, which is a novel definition developed from the concept of ergodic matrix, is presented to describe the pixel position shifting between the original image and the corresponding encrypted image. Any element in a fuzzy ergodic matrix is a set, which means that mapping can be multiplex. Furthermore, we define an operation system of fuzzy ergodic matrices, including intersection, union, clearance, etc.

Concept and definition

(1) Fuzzy ergodic matrix

An $m\times n$ matrix $\tilde{\boldsymbol{R}}_{m\times n}=\{\tilde{r}(i,j):1\le i\le m,$ $1\le j\le n\}$ is defined as a fuzzy ergodic matrix, if every element of which is a subset of $\{1, 2, \ldots, mn\}$ and satisfy the following conditions:

C1: $\tilde{r}(i,j)\neq\Phi$

C2: For any two elements $\tilde{r}(i_1,j_1)$ and $\tilde{r}(i_2,j_2)$, there must be $\tilde{r}(i_1,j_1)\cap\tilde{r}(i_2,j_2)=\Phi$, or $\tilde{r}(i_1,j_1)\cap\tilde{r}(i_2,j_2)=\tilde{r}(i_1,j_1)=\tilde{r}(i_2,j_2)$

C3: $\bigcup\limits_{\substack{1\le i\le m\\1\le j\le n}}\tilde{r}(i,j)=\{1,\ldots,mn\}$

C4: The time that $\tilde{r}(i,j)$ emerges in the matrix is equal to $\|\tilde{r}(i,j)\|$ (the number of elements in the set $\tilde{r}(i,j)$)

(2) Fixed rate/fixed position

$$\textit{fixed position}=\textit{position that satisfy }\|\tilde{r}(i,j)\|=1$$

$$\textit{fixed rate}=\frac{\textit{number of fixed position}}{mn}$$

(3) Definition rate/matrix scale

The number of elements in all the sets in $\tilde{\boldsymbol{R}}$ represents the memory size needed. The higher the number is, the larger the memory size needed, in other words, the greater the computational complexity will be. Definitions of definition scale and definition rate are described below:

$$\textit{matrix scale}=\textit{total elements number of all the sets in }\tilde{\boldsymbol{R}}$$

$$\textit{definition rate}=\frac{mn}{\textit{matrix scale}}$$

(4) Decryption probability/decryption space

It is believed that $\tilde{\boldsymbol{R}}$ could be transformed into $\boldsymbol{R}$ and that the probability is related to the fixed rate. Decryption space and decryption probability are defined as follows:

$$\textit{decryption space}=\textit{number of }\boldsymbol{R}\textit{ could be transformed from }\tilde{\boldsymbol{R}}$$

$$\textit{decryption probability}=\frac{1}{\textit{decryption space}}$$

Operation system

Although the fuzzy ergodic matrix has been defined, the corresponding operation system, which is useful in practice, remains to be identified. For example, a sort of “intersection” operation will be needed. And the “clearance” operation is necessary to transform a fuzzy ergodic matrix into an ergodic matrix.

(1) Intersection

For two $m\times n$ fuzzy ergodic matrices $\tilde{\boldsymbol{R}}_1$ and $\tilde{\boldsymbol{R}}_2$, we define the intersection of $\tilde{\boldsymbol{R}}_1$ and $\tilde{\boldsymbol{R}}_2$ as the intersection of arbitrary element in $\tilde{\boldsymbol{R}}_1$ with the corresponding element in the same position in $\tilde{\boldsymbol{R}}_2$. The intersection is still an $m\times n$ fuzzy ergodic matrix, as presented below:

$$\tilde{\boldsymbol{R}}_1 \cap \tilde{\boldsymbol{R}}_2 = \begin{bmatrix} R_1(1,1) & R_1(1,2) & \cdots & R_1(1,n) \\ R_1(2,1) & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ R_1(m,1) & \cdots & \cdots & R_1(m,n) \end{bmatrix} \cap \begin{bmatrix} R_2(1,1) & R_2(1,2) & \cdots & R_2(1,n) \\ R_2(2,1) & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ R_2(m,1) & \cdots & \cdots & R_2(m,n) \end{bmatrix}$$

$$= \begin{bmatrix} R_1(1,1)\cap R_2(1,1) & R_1(1,2)\cap R_2(1,2) & \cdots & R_1(1,n)\cap R_2(1,n) \\ R_1(2,1)\cap R_2(2,1) & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ R_1(m,1)\cap R_2(m,1) & \cdots & \cdots & R_1(m,n)\cap R_2(m,n) \end{bmatrix}$$

Example:

$$\tilde{\boldsymbol{R}}_1 = \begin{bmatrix} 8 & 7 & (4,6,9) \\ (4,6,9) & 3 & (4,6,9) \\ (1,5) & 2 & (1,5) \end{bmatrix}$$

$$\tilde{\boldsymbol{R}}_2 = \begin{bmatrix} (8,7,2) & (8,7,2) & (6,9) \\ (6,9) & (3,5) & 4 \\ 1 & (8,7,2) & (3,5) \end{bmatrix}$$

Then,

$$\tilde{\boldsymbol{R}} = \tilde{\boldsymbol{R}}_1 \cap \tilde{\boldsymbol{R}}_2 = \begin{bmatrix} 8 & 7 & (4,6,9) \\ (4,6,9) & 3 & (4,6,9) \\ (1,5) & 2 & (1,5) \end{bmatrix} \cap \begin{bmatrix} (8,7,2) & (8,7,2) & (6,9) \\ (6,9) & (3,5) & 4 \\ 1 & (8,7,2) & (3,5) \end{bmatrix}$$

$$= \begin{bmatrix} 8\cap(8,7,2) & 7\cap(8,7,2) & (4,6,9)\cap(6,9) \\ (4,6,9)\cap(6,9) & 3\cap(3,5) & (4,6,9)\cap 4 \\ (1,5)\cap 1 & 2\cap(8,7,2) & (1,5)\cap(3,5) \end{bmatrix}$$

$$= \begin{bmatrix} 8 & 7 & (6,9) \\ (6,9) & 3 & 4 \\ 1 & 2 & 5 \end{bmatrix}$$

Similarly we define:

$$\tilde{\boldsymbol{R}}_1 \cap \tilde{\boldsymbol{R}}_2 \cap \tilde{\boldsymbol{R}}_3 = \tilde{\boldsymbol{R}}_1 \cap (\tilde{\boldsymbol{R}}_2 \cap \tilde{\boldsymbol{R}}_3)$$

Thus we have:

$$\tilde{\boldsymbol{R}}_1 \cap \tilde{\boldsymbol{R}}_2 \cap \cdots \cap \tilde{\boldsymbol{R}}_k,\ k=2,3,\ldots$$

Clearance

For image encryption and decryption, the fuzzy ergodic matrix has no practical value by itself. But a corresponding ergodic matrix can be generated from it. Because the informations are not enough, and all the position information is uncertain, a clearance should be made:

$$\tilde{\boldsymbol{R}} = \begin{bmatrix} \tilde{R}(1,1) & \cdots & \tilde{R}(1,n) \\ \cdots & \cdots & \cdots \\ \tilde{R}(m,1) & \cdots & \tilde{R}(m,n) \end{bmatrix} \xrightarrow{clearance}$$

$$\boldsymbol{R} = \begin{bmatrix} R(1,1) & \cdots & R(1,n) \\ \cdots & \cdots & \cdots \\ R(m,1) & \cdots & R(m,n) \end{bmatrix}$$

Using an ergodic pattern (for example row ergodic pattern) (Zhao and Chen, 2002), the minimal values of the set elements in $\tilde{\boldsymbol{R}}(i,j)$ are each taken in turn as the value of $\boldsymbol{R}(i,j)$ (we name this method as the "min value method"), and thus get $\tilde{\boldsymbol{R}}$ clear.

Example:

$$\tilde{\boldsymbol{R}} = \begin{bmatrix} 8 & 7 & (4,6,9) \\ (4,6,9) & 3 & (4,6,9) \\ (1,5) & 2 & (1,5) \end{bmatrix} \xrightarrow{1} \begin{bmatrix} 8 & 7 & 4 \\ (6,9) & 3 & (6,9) \\ (1,5) & 2 & (1,5) \end{bmatrix}$$

$$\xrightarrow{2} \begin{bmatrix} 8 & 7 & 4 \\ 6 & 3 & 9 \\ (1,5) & 2 & (1,5) \end{bmatrix} \xrightarrow{3} \begin{bmatrix} 8 & 7 & 4 \\ 6 & 3 & 9 \\ 1 & 2 & 5 \end{bmatrix} = \boldsymbol{R}$$

DECRYPTION OF PURE-POSITION ALGORITHMS

Probability theory can be used to prove that the image encrypted by pure position permutation algorithm can be decrypted if we obtain a lot of original images and their corresponding encrypted ones as reference.

If there are k image pairs, whose scale is $m\times n$ and the histogram set is $G=\{0, 1, \ldots, L\}$, then all position sets of the image matrix can be presented as $\Omega=\{(1, 1), \ldots, (m, n)\}=\{1, 2, \ldots, mn\}$.

Definition of encryption pairs

(1) Source images

$$\boldsymbol{I}_1=\begin{bmatrix} I_1(1,1) & \cdots & I_1(1,n) \\ \cdots & \cdots & \cdots \\ I_1(m,1) & \cdots & I_1(m,n) \end{bmatrix}_{m\times n}, I_2, \ldots, I_k$$

(2) Corresponding encrypted images

$$\boldsymbol{E}_1=\begin{bmatrix} E_1(1,1) & \cdots & E_1(1,n) \\ \cdots & \cdots & \cdots \\ E_1(m,1) & \cdots & E_1(m,n) \end{bmatrix}_{m\times n}, E_2, \ldots, E_k$$

(3) Other encrypted images (without the corresponding source images)

$$\boldsymbol{E}_{k+1}=\begin{bmatrix} E_1(1,1) & \cdots & E_1(1,n) \\ \cdots & \cdots & \cdots \\ E_1(m,1) & \cdots & E_1(m,n) \end{bmatrix}_{m\times n}, E_{k+2}, \ldots, E_{k+p}$$

Generate fuzzy ergodic matrix

The above discussion defined the fuzzy ergodic matrix $\tilde{\boldsymbol{R}}$; now it is used to note the positional relation among the image pairs. To do this, we should identify all the positions whose values equal to $E(i, j)$ should be identified and indicated in the corresponding positions of $\tilde{\boldsymbol{R}}(i, j)$.

Example:

$$\boldsymbol{I}_{m\times n}=\begin{bmatrix} A & E & F \\ B & A & B \\ C & D & B \end{bmatrix} \xrightarrow{f_R} \boldsymbol{E}_{m\times n}=\begin{bmatrix} D & C & B \\ B & F & B \\ A & E & A \end{bmatrix}$$

With $I_{i\times n+j}=I(i, j)$, then

$I_1=A$, $I_2=E$, $I_3=F$, $I_4=B$, $I_5=A$, $I_6=B$, $I_7=C$, $I_8=D$, $I_9=B$,

and

$E_1=D$, $E_2=C$, $E_3=B$, $E_4=B$, $E_5=F$, $E_6=B$, $E_7=A$, $E_8=E$, $E_9=A$,

Thus, $\tilde{\boldsymbol{R}}$ can be generated as follows:

$$\tilde{\boldsymbol{R}}=\begin{bmatrix} \{8\} & \{7\} & \{4,6,9\} \\ \{4,6,9\} & 3 & \{4,6,9\} \\ \{1,5\} & 2 & \{1,5\} \end{bmatrix}$$

SIMULATION RESULTS AND CONCLUSIONS

A series of gray 256-scales images of size 128×128 were selected and encrypted using a certain pure position algorithm (random-matrix change). Then, some image reference pairs were used to decrypt the encrypted image. The results were satisfactory.

A series of encryption image pairs (Fig.1)

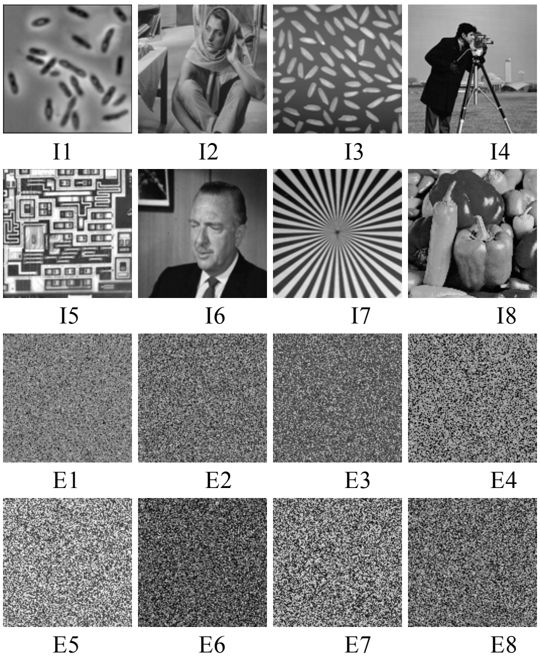

Fig.1 A series of original images and corresponding encrypted images

Decryption result

The recovered images are shown as in Fig. 2−Fig.6.

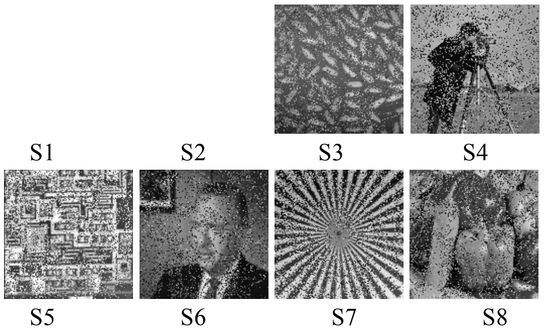

Fig.2 Recovered images (*k*=2)

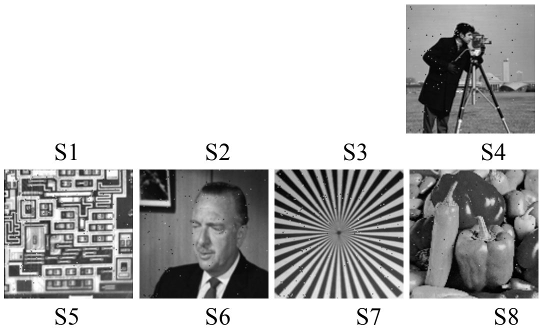

Fig.3 Recovered images (*k*=3)

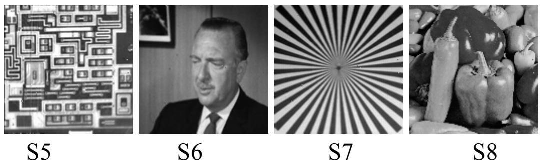

Fig.4 Recovered images (*k*=4)

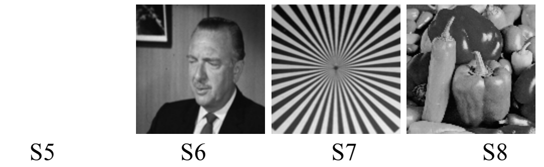

Fig.5 Recovered images (*k*=5)

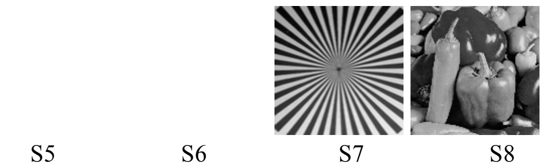

Fig.6 Recovered images (*k*=6)

When *k*=2, the encrypted images (E3~E8) were decrypted by using the first two image pairs (I1-E1, I2-E2); the fixed rate was 39.843750%; the definition rate was 44.226097% and the decryption probability was 0.000000%.

When *k*=3, the encrypted images (E3~E8) were decrypted by using the first two image pairs (I1-E1, I2-E2), the fixed rate was 96.539307%, the definition rate was 96.376471% and the decryption probability was 0.000000%.

When *k*=4, the encrypted images (E5~E8) were decrypted by using the first four image pairs (I1-E1, I2-E2, I3-E3, I4-E4), the fixed rate was 99.639893%, the definition rate was 99.623009% and the decryption probability was 0.000000%.

When *k*=5, the encrypted images (E6~E8) were decrypted by using the first five image pairs (I1-E1, I2-E2, I3-E3, I4-E4, I5-E5), the fixed rate was 99.987793%, the definition rate was 99.987794% and the decryption probability was 50%.

When *k*=6, the encrypted images (E7~E8) were decrypted by using the first six image pairs (I1-E1, I2-E2, I3-E3, I4-E4, I5-E5, I6-E6), the fixed rate was 100%; the definition rate was 100% and the decryption probability was 100%.

In this test, we could decrypt completely any encrypted images using this pure-position permutation algorithm after the sixth step. In fact, we tested over 1000 gray 256-scales image pairs (from size 64×64 to 512×512). Almost all of the test results were satisfyingly (limited to 6 steps).

There were only about 10 exceptions in all the over 1000 image pairs. Analysis of the characteristic of the exceptions showed that the distribution of the pixels' gray values were terribly unbalanced in these worst cases. It was obvious that higher probability of the repetition of gray values led to greater difficulty in the decryption.

Probability theory and algebraic principle can be used to prove that the decryption of pure-position permutation algorithms is feasible theoretically.

References

Bourbakis, N., Alexopoulos, C., 1992. Picture data encryption using SCAN patter. *Pattern Recognit.*, **25**(6): 567-581.

Ding, W., Qi, D.X., 1998. Digital image transformation and information hiding and disguising technology. *Chinese Journal of Computer*, **21**(9):838-843 (in Chinese).

Kuo, C.J., Chen, M.S., 1991. A New Signal Encryption Technique and its Attack Study. Proceedings of IEEE International Conference on Security Technology, Taipei, Taiwan, **l**:149-153.

Lin , T.Q., Klara, N., 1998. Camparison of MPEG encryption algorithms. *Comput. & Graphics*, **22**(4):437-448.

Matlas, Y., Shamir, A., 1992. A video scrambling technique based on space filling curves. *Proceedings of CPYPTO'87*, **76**(5):550-559.

Refregier, P., Javidi, B., 1995. Optical-image encryption based on input plane and forier plane random encoding. *Optics Letters*, **20**(7):767-769.

Sridharan, S., Dawson, E., Goldburg, B., 1991. Fast Fourier transform based speech encryption system. *IEE Proc. I, Commun. Speech Vis*, **138**(3):215-223.

Yang, H.G., Kim, E.S., 1996. Practical image encryption scheme by real-valued data. *Opt. Eng.*, **35**(9):2473-2478.

Yen, J.C., Guo, J.I., 1999. A Chaotic Neural Network for Signal Encryption/Decryption and its VLSTI Architecture. Proceedings of the Tenth VLSI Design/CAD Symposium, Nantou, Taiwan, p.319-322.

Zhao, X.Y., Chen, G., 2002. Ergodic matrix in image encryption. *SPIE*, p.4875-4878.