A Novel Color Image Encryption Scheme Based on Permutation-substitution Architecture

Rui-Song Ye¹,a, Ming Ye¹, Xiao-Yun Shi¹, Wen-Hao Ye¹, Ya-Fang Li¹

¹Department of Mathematics, Shantou University, Shantou, Guangdong, 515063, China

Keywords: Skew tent map, Permutation, Substitution, Image encryption.

Abstract. A novel color image encryption scheme with permutation-substitution architecture is proposed. One round of permutation and one round of substitution achieve desirable results. The color plain-image sized $H \times W$ is converted to a 2D matrix with size $H \times 3W$ and then divided into two equal parts $P_1, P_2$. The 2D skew tent map is applied to generate pseudo-random sequences for the permutation and substitution processes implemented row-by-row/column-by-column instead of pixel-by-pixel to increase the encryption rate. The permutation is performed between $P_1, P_2$ and the substitution is executed within the permuted image. The security and performance of the proposed scheme have been analyzed as well. All the experimental results show that the proposed scheme is secure and effective for practical application.

Introduction

Thanks to the fantastic features of chaotic systems, such as high sensitivity to initial conditions and control parameters, ergodicity, pseudo-randomness etc., chaos-based image encryption schemes are extensively studied and developed recently. On one hand, traditional encryption algorithms, such as DES, AES, are typically designed for textual information and are not suitable for image encryption due to the intrinsic natures of images like high redundancy and high correlation among pixels [1]. On the other hand, the good chaotic natures agree with the fundamental requirements such as confusion and diffusion in cryptography, and therefore chaotic systems provide a potential candidate for constructing cryptosystems [2-4].

Since Fridrich firstly presented the fundamental permutation-diffusion architecture of chaos-based image encryption in 1998, a great number of chaos-based image encryption algorithms with such permutation-diffusion architecture have been proposed [2-6]. However, Wang et al. found that the typical permutation-diffusion architecture with fixed parameters has one fatal drawback, that is, the two processes will become independent if the plain-image is a homogeneous one with identical pixel gray value [7]. As a matter of fact, some image encryption algorithms with permutation-diffusion architecture have been broken by chosen-plaintext or known-plaintext attacks [8,9].

In this paper, we present an image encryption scheme with permutation-substitution mechanism instead of permutation-diffusion mechanism. Permutation-substitution mechanism has been shown to be one effective mechanism for constructing ciphers [6]. The image encryption scheme proposed here consists of two stages: one permutation and one substitution. They are performed between the $R, G, B$ components of color image. The color plain-image sized $H \times W$ is converted to a 2D matrix with size $H \times 3W$ and then is divided into two parts.
1, 2

with the same size \( H \times 1.5W \). To achieve desirable key sensitivity and plaintext sensitivity, the permutation stage is designed to be dependent on plain-image. As a result, the proposed image scheme owns good resistance to known-plaintext and chosen-plaintext attacks. The substitution stage is performed row by row and column by column to improve the encryption rate. The substitution achieves good diffusion effect and shows good resistance against differential analysis as well. The security and performance analysis of the proposed image encryption are carried out thoroughly. All the experimental results show that the proposed image encryption scheme is highly secure and demonstrates excellent performance.

The Proposed Image Encryption Scheme

Read a color plain-image \( PI \) expressed by a 3D matrix with size \( H \times W \times 3 \). We assume that the three color components are denoted by 2D matrix \( R, G, B \) respectively. Then convert \( PI \) to a 2D matrix \( P \) with size \( H \times 3W \) by the way \( P = [R, G, B] \). \( P \) is then divided into two parts \( P_1, P_2 \), whose sizes are the same \( H \times 1.5W \). For the sake of simplicity, we assume that \( W \) is even and let \( W_1 = 1.5W \), \( T = \max(H, W_1) \). The conversion and division is shown in Fig. 1.

The proposed image encryption scheme is composed of one permutation process and one substitution process. The entire image encryption scheme is outlined as follows.

Step 1. Generation of pseudo-random gray value vectors \( IVR, IVC \). With cipher keys \( x_0, y_0, a, b \) and \( N \), we iterate the 2D skew tent map for \( N \) times and reject the transient points \( \{(x_k, y_k) : k = 0, 1, \ldots, N-1\} \). The values of \( (x_k, y_k) \) are saved and the 2D skew tent map with new initial values \( (x_k, y_k) \) to yield \( IVR, IVC \). We still write \( (x_k, y_k) \) as \( (x_0, y_0) \).

\[
\begin{align*}
\left( x_i \right) &= \begin{cases} 
\frac{x_{i-1}}{p}, & \text{if } x_{i-1} \in [0, p), \\
 \frac{1-x_{i-1}}{1-p}, & \text{if } x_{i-1} \in (p, 1], 
\end{cases} \\
\left( y_i \right) &= \text{floor}(x_i \times 256), \text{ floor}(y_i \times 256), \ \text{for } i = 1, \ldots, T,
\end{align*}
\]

where \( \text{floor}(x) \) returns the largest integer not larger than \( x \). Truncate the first \( H \) elements of \( IVC \) and transpose it to get one column vector \( IVR \) with \( H \) elements. Truncate the first \( W_1 \) elements of \( IVR \) to get one row vector \( IVR \) with \( W_1 \) elements.

Step 2. Generation of pseudo-random gray value vectors \( SVR, SVC \). We still denote \( (x_i, y_i) \) as \( (x_0, y_0) \) for simplicity.

\[
\begin{align*}
\left( x_i \right) &= \begin{cases} 
\frac{x_{i-1}}{p}, & \text{if } x_{i-1} \in [0, p), \\
 \frac{1-x_{i-1}}{1-p}, & \text{if } x_{i-1} \in (p, 1], 
\end{cases} \\
\left( y_i \right) &= \text{floor}(x_i \times 256), \text{ floor}(y_i \times 256), \ \text{for } i = 1, \ldots, T,
\end{align*}
\]

Step 3. Perform the permutation stage. Calculate the number of iterations to skip before starting the permutation by \( N_i = P(1,1) + \ldots + P(1,3W) + P(2,1) + \ldots + P(H,3W) \mod 256 \). Starting
with the initial conditions \((x_0, y_0)\) obtained in Step 1, we iterate the 2D skew tent map for \(N_1\) times and save the new values \((x_{N_1}, y_{N_1})\) as \((x_0, y_0)\). For \(i = 1\) to \(T\), do the following loop

\[
\begin{cases}
  x_i = \frac{x_{i-1}}{p}, & \text{if } x_{i-1} \in [0, p], \\
  (1-x_{i-1})/(1-p), & \text{if } x_{i-1} \in (p, 1],
\end{cases}
\]

\(PR(i) = \text{floor}(x \cdot H) + 1, PC(i) = \text{floor}(y \cdot W1) + 1.\)

The vectors \(PR, PC\) are then employed to perform the permutation between \(P1\) and \(P2\) row-by-row and column-by-column via the following loop and get one permuted image \(G1\). For \(j = 1\) to \(H\), exchange the \(PR(j)\)-th row of \(P2\) with the \(j\)-th row of \(P1\); For \(j = 1\) to \(W1\), exchange the \(PC(j)\)-th column of \(P2\) with the \(j\)-th column of \(P1\).

Step 4. Substitute the 2D matrix \(G1\) row-by-row and column-by-column. The execution for the substitution is defined by

\[
G1(i,:) = G1(i,:) \oplus IVR \oplus SVR(1); \quad G1(i,:) = G1(i,:) \oplus G1(i-1,:) \oplus SVR(i), \quad i = 2, \ldots, H,
\]

\[
G1(:,1) = G1(:,1) \oplus IVC \oplus SVC(1); \quad G1(:,j) = G1(:,j) \oplus G1(:,j-1) \oplus SVC(j), \quad j = 2, \ldots, W1,
\]

where “\(\oplus\)” represents the bitwise XOR operation, and \(G1(i,:), G1(:,j)\) denote the \(i\)-th row and \(j\)-th column of \(G1\). The resulted cipher-image for plain-image Lena is shown in Fig. 2(b).

Note: (a) plain-image Lena; (b) cipher-image; (c),(d),(e): histograms for \(R,G,B\) components of Lena; (f),(g),(h): histograms for \(R,G,B\) components of cipher-image.

Fig. 2. The encrypted results.

**Performance Analysis**

**Histogram analysis**

Histogram analysis is a visual test which shows the pixel distribution over the available intensity levels. For a 24-bit color image, three histograms can be drawn for each 8-bit red, green and blue channel. Encrypt the color image Lena one round with cipher keys \((x_0, y_0, a, b, N)\) to be \((0.20, 0.44, 0.23, 0.57, 933)\), and then plot the histograms of plain-image
and cipher-image as shown in Fig. 2. One can conclude from the histograms of the cipher-image that they are fairly uniform and significantly different from the corresponding histograms of the plain-image.

**Correlation coefficient analysis**

It is common sense that for one nature image with definite visual contents, each pixel is highly correlated with its adjacent pixels either in horizontal, vertical or diagonal direction. An ideal encryption cryptosystem should produce cipher-images with less correlation in the adjacent pixels. We select $T = 6000$ pairs of two adjacent pixels randomly from an image and then calculate the correlation coefficient of the selected pairs by $Cr = \frac{cov(x,y)}{\sqrt{D(x)D(y)}}$, $cov(x,y) = \frac{1}{T} \sum_{i=1}^{T} (x_i - E(x))(y_i - E(y))$, $E(x) = \frac{1}{T} \sum_{i=1}^{T} x_i$, $D(x) = \frac{1}{T} \sum_{i=1}^{T} (x_i - E(x))^2$, where $x_i, y_i$ form the $i$-th pair of horizontally, vertically or diagonally adjacent pixels. The correlation coefficients of horizontally, vertically, diagonally adjacent pixels for plain-image Lena and its cipher-image are given in Table 1. It is clear from Table 1 that the proposed image encryption scheme significantly reduces the correlation between the adjacent pixels of the plain-image.

**Information entropy analysis**

Information entropy measures the disorder and randomness of information sequence [10]. Regarding image, it can be used to measure the uniformity of image histograms. The entropy $H(m)$ of a message source $m$ can be calculated by $H(m) = -\sum_{i=0}^{L-1} p(m_i) \log(p(m_i))$ (bits), where $L$ is the total number of symbols $m$, $p(m_i)$ represents the probability of occurrence of symbol $m_i$, and log denotes the base 2 logarithm so that the entropy is expressed in bits. For a 24-bit color image, the information entropy for each color channel (Red, Green and Blue) is given as

$$H_{R/G/B}(m) = \sum_{i=0}^{2^{24}-1} p_{R/G/B}(RI_i) \log_2 \frac{1}{p_{R/G/B}(RI_i)} \text{ (bits).}$$

**Tab. 1. Correlation coefficients between adjacent pixels of plain and cipher image.**

<table>
<thead>
<tr>
<th></th>
<th>Correlation between adjacent pixels</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Red</td>
</tr>
<tr>
<td><strong>Horizontal</strong></td>
<td></td>
</tr>
<tr>
<td>Plain-image</td>
<td>0.9762</td>
</tr>
<tr>
<td>Cipher-image</td>
<td>0.0229</td>
</tr>
<tr>
<td><strong>Vertical</strong></td>
<td></td>
</tr>
<tr>
<td>Plain-image</td>
<td>0.9864</td>
</tr>
<tr>
<td>Cipher-image</td>
<td>0.0334</td>
</tr>
<tr>
<td><strong>Diagonal</strong></td>
<td></td>
</tr>
<tr>
<td>Plain-image</td>
<td>0.9622</td>
</tr>
<tr>
<td>Cipher-image</td>
<td>0.0473</td>
</tr>
</tbody>
</table>

We have calculated the information entropy for plain-image Lena and its cipher image. The results are shown in Table 2. The value of information entropy for the cipher-image is very-very close to the expected value of truly random image, i.e., 8bits. Hence the proposed encryption scheme is extremely robust against entropy attacks.
Tab. 2. Information entropy analysis.

<table>
<thead>
<tr>
<th></th>
<th>Red</th>
<th>Green</th>
<th>Blue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plain-image</td>
<td>7.263435</td>
<td>7.589897</td>
<td>6.985354</td>
</tr>
<tr>
<td>Cipher-image</td>
<td>7.999300</td>
<td>7.999300</td>
<td>7.999363</td>
</tr>
</tbody>
</table>

Differential attack analysis

Differential cryptanalysis is the study of how differences in a plaintext can affect the resultant differences in the ciphertext with the same cipher key. If one slight difference in the plain-image will cause significant, random and unpredictable changes in the cipher-image, then the encryption scheme will resist differential analysis attack efficiently. Two most common measures NPCR (number of pixel change rate) and UACI (unified average changing intensity) are used to test the robustness of image cryptosystems against the differential cryptanalysis. If $C_{R/G/B}^{k_1}$ and $C_{R/G/B}^{k_2}$ represent the $R, G, B$ channels for two cipher-images, then NPCR for each color channel is defined by

$$\text{NPCR}_{R/G/B} = \frac{\sum_{i=1}^{H} \sum_{j=1}^{W} D_{i,j}^{R/G/B}}{W \times H} \times 100\%,$$

where $D_{i,j}^{R/G/B} = \begin{cases} 0, & \text{if } C_{i,j}^{R/G/B} = C_{i,j}^{k_1} \\ 1, & \text{if } C_{i,j}^{R/G/B} \neq C_{i,j}^{k_1} \end{cases}.$

UACI is calculated by

$$\text{UACI}_{R/G/B} = \frac{1}{W \times H} \sum_{i=1}^{H} \sum_{j=1}^{W} \frac{|C_{i,j}^{R/G/B} - C_{i,j}^{k_1}|}{2^{255/8}} \times 100\%.$$

We have performed the differential analysis by calculating NPCR and UACI on plain-image Lena. The analysis has been done by randomly choosing 500 pixels in plain-image, and changing all three color intensity values by one unit at the selective pixel. The averages of 500 NPCR values and 500 UACI values thus obtained for all three color components are given in Table 3. It is clear that the NPCR and UACI values are very close to the expected values, thus the proposed image encryption technique shows good sensitivity to plaintext and hence invulnerable to differential attacks.

Tab. 3. Difference analysis of plain-image Lena.

<table>
<thead>
<tr>
<th></th>
<th>Average NPCR (%)</th>
<th>Average UACI (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Red</td>
<td>Green</td>
</tr>
<tr>
<td>Lena</td>
<td>99.5819</td>
<td>99.6185</td>
</tr>
</tbody>
</table>

Key sensitivity analysis

A good image encryption scheme should be extremely sensitive to cipher keys, which is an essential feature for any good cryptosystem in the sense that it can effectively prevent invaders decrypting original data. If one uses two slightly different keys to encrypt the same plain-image, then two cipher-images should possess negligible correlation. The plain-image is respectively encrypted with one master cipher key and five other cipher keys which have only a minor difference in any one of five parts of master cipher key. Six cipher keys are used to encrypt image Lena. Master cipher key is $\text{MKEY}(0.20, 0.44, 0.23, 0.57, 933)$; five slightly different
keys are $\text{SKEY1}(0.20-10^{14},0.44,0.23,0.57,933)$, $\text{SKEY2}(0.20,0.44-10^{14},0.23,0.57,933)$, $\text{SKEY3}(0.20,0.44,0.23-10^{14},0.657,933)$, $\text{SKEY4}(0.20,0.44,0.23,0.57-10^{14},933)$, $\text{SKEY5}(0.20,0.44,0.23,0.57,933+1)$.

We then calculate the 2D correlation coefficients between the various color channels of the cipher-image by MKEY and five other cipher-images by SKEY1,...,SKEY5. The results are provided in Table 4. All the correlation coefficients are very small or practically zero indicating that all the cipher-images are highly different.

| Correlation coefficients between the cipher-images obtained using MKEY and SKEY1 SKEY2 SKEY3 SKEY4 SKEY5 |
|---------------------------------|---------------------------------|---------------------------------|---------------------------------|---------------------------------|
| Crr 0.002632377 | 0.000044931 | 0.000436494 | -0.001905759 | -0.000075756 |
| Crg -0.000057923 | -0.001377704 | -0.000783237 | 0.002244601 | -0.0000464126 |
| Crb 0.003348934 | 0.003703526 | -0.000073924 | 0.000070452 | -0.003745186 |
| Cgr -0.005103932 | 0.000601544 | 0.000908763 | -0.005642771 | 0.0000040334 |
| Cgb -0.003346569 | -0.002904194 | 0.000964266 | -0.003590179 | 0.00195766 |
| Cbr 0.001743543 | 0.003639789 | 0.001375066 | 0.001225237 | -0.000709332 |
| Cbg -0.002084912 | -0.001186359 | 0.001394259 | -0.001394872 | -0.002179607 |
| Cbb 0.002632377 | 0.000044931 | 0.000436494 | -0.001905759 | -0.000075756 |

Acknowledgement

This research is supported by the Challenge Promotion Plan of Shantou University.

References


