基于三维分割的图像加密算法

农盛功，周满元

(桂林电子科技大学 计算机与控制学院，广西 桂林 541004)

摘 要：为了实现图像安全、快速加密，设计了一种新的数字图像混沌加密算法。仿真结果表明，该算法在没有扩散函数的情况下也能很好地改变图像的直方图，可以有效地抵制已知(选择) 明文攻击，而且只有 1 次左映射或者右映射的情况下，该方法具有高置乱程度、良好的扩散性能和足够大的密钥空间，可获得足够高的安全性。

关键词：混沌；映射；置乱

中图分类号：TP309.7 **文献标识码**：A

Image encryption based on the segmentation of three-dimensional space

NONG Sheng Gong, ZHOU Man Yuan

(College of Computer Science and Control, Guilin University of Electronic Technology, Guilin 541004, China)

Abstract: In order to satisfy the need of security and speed of image encryption, this paper proposes a novel chaotic encryption about digital image. The simulation results show that, without the diffusion-algorithm, it can change the histogram of image and resist the plain-text attack very well. Furthermore, even though you just only use once left map or once right map, this algorithm has high confusion , good diffusion, large enough key space and high secure.

Key words: chaotic; mapping; compound

随着 Internet、多媒体技术、信息存储技术的迅速发展以及网络带宽限制的放宽，越来越多的数字图像在网络上传输，成为人们获取信息的重要手段之一。因此对发送的图像进行可靠的安全处理也成为当前研究的重要方向之一[1]。

混沌系统本身具有伪随机性、对系统参数和初始值的敏感性、奇异吸引子、难以分析性的特性，所以混沌系统可以提供大量的具有良好随机性、非相关性和复杂性的伪随机序列，非常适用于图像加密。混沌加密主要分为置乱加密技术[2]、置换加密技术[3]以及置乱和置换相结合的加密技术。置乱加密技术不会引起信息的冗余，尤其适合大幅图像的加密。图像置乱不能改变图像的像素值，它不能有效地抵制统计攻击。而图像置换就是为了抵制统计攻击而设计的，其目的就是降低相邻像素之间的相关性，增加图像的紊乱程度，从而使加密图像中所含的信息难以统计而达到抵制统计攻击的目的。

1 二维十进制矩阵转化为三维二进制矩阵

数字图像的处理一般基于二维空间，本文提出的算法是在三维空间中实现的，因此首先要把二维图像读取成为三维图像，具体方法见参考文献[4]。

2 映射思想和算法

2.1 映射思想

设三维矩阵在 IJ 坐标上投影成一个 $N\times N$ 二维矩阵(当投影不是 $N\times N$ 的方阵时，可以用插补法把它补成方阵)，映射方法就是根据图像像素能够插入到其他相邻的像素之间的性质。根据方向的不同，映射分为左映射和右映射。首先利用平面 $I=J$ 或者平面 $J=N-I$，即在 IJ 平面投影中的方图的主对角线或者反对角线，且平行于 K 轴的平面，将图像分割成 2 个方块。方块在 IJ 平面上的投影都是等腰三角形，由于等腰三角形图像每一列的二进制数目和相邻列的二进制数目是不同的，因此可以沿着水平方向分别把平行于 K 轴的每一列中的所有二进制数分别插入到相邻的二进制数之间。反复重复该过程，依次链接，则原始 $8\times N\times N$ 的三维矩阵被拉伸成为一个 $8\times(N\times N)$ 的二维矩阵，然后用二维矩阵置乱算法将 $8\times(N\times N)$ 的二维矩阵置乱，再将其还原成为 $8\times N\times N$ 的三维矩阵。如此完成一次映射过程。图 1、图 2、图 3 为左映射的各个过程。

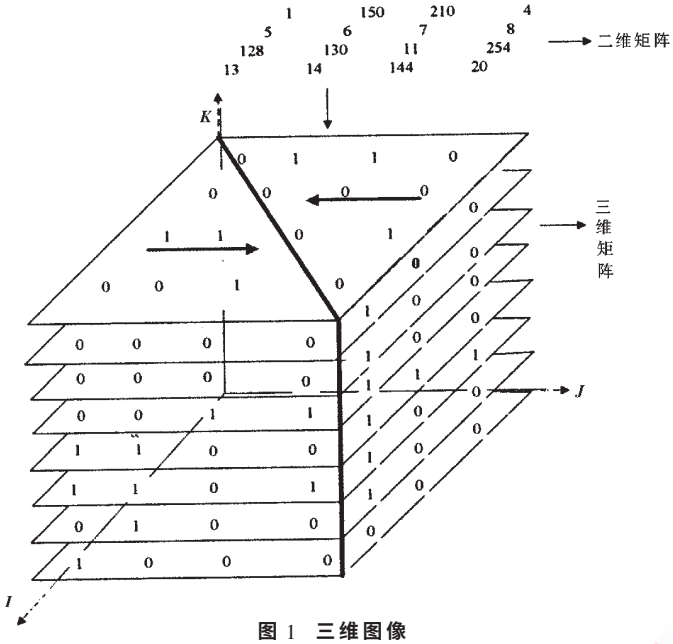

图 1 三维图像

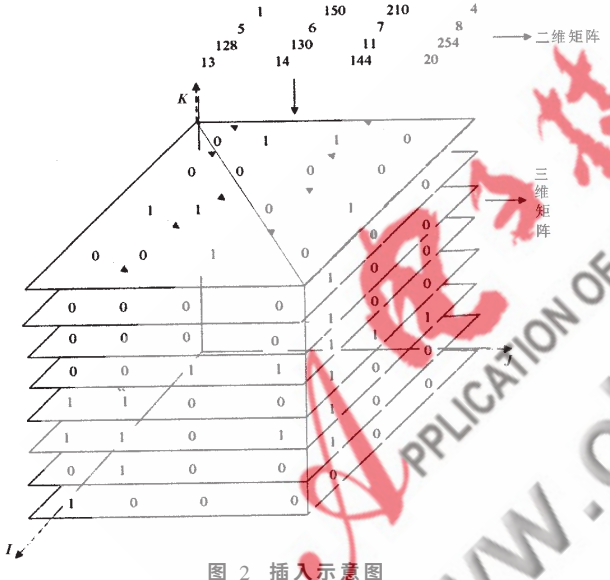

图 2 插入示意图

0	0	1	0	0	1	0	0	0	1	0	0	1	1	0	1
0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	1	0	0	1	0	0	0	1	0	0	0	0	0	1
0	1	1	0	1	0	0	0	0	0	1	1	0	0	0	0
1	0	1	1	0	0	1	1	0	1	1	1	0	1	1	0
0	1	1	1	0	1	0	1	0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0	1	0	1	0	0	1	1	0

图 3 拉伸成的二维矩阵

当映射为右映射时，插入方向正好与左映射相反，链接方向也相反，右映射与左映射对称。

2.2 计算方法

设图像大小为 $N\times N$，读成三维矩阵后为 $\boldsymbol{A}(k,i,j)$ $(k=0,1,\cdots,7;\ i=0,1,\cdots,N-1;\ j=0,1,\cdots,N-1)$。设 $\boldsymbol{P}(k,m)$ 为拉伸成一个平面后的二维矩阵，其中 $k=0,1,\cdots,7;m=0,1,\cdots,N\times N-1$。（下文中 k、i、j、m 取值范围同上）

左映射计算方法如下（下文式中[]为取整符号）：

当 $j\geqslant i$ 且 $N-j$ 是奇数时

$$\boldsymbol{P}\left(k,\left[\frac{(N+j+2)(N-j-1)}{2}+2(j-i)\right]\right)=\boldsymbol{A}(k,i,j) \tag{1}$$

当 $j\geqslant i$ 且 $N-j$ 是偶数时

$$\boldsymbol{P}\left(k,\left[\frac{(N+j+3)(N-j-2)}{2}+2(j-i)+1\right]\right)=\boldsymbol{A}(k,i,j) \tag{2}$$

当 $j<i$ 且 j 是偶数时

$$\boldsymbol{P}\left(k,\left[\frac{(N^2+N+(2N-j-1)\times j}{2}+2(N-i-1)\right]\right)=\boldsymbol{A}(k,i,j) \tag{3}$$

当 $j<i$ 且 j 是奇数时

$$P\left(k,\left[\frac{(N^2+N+(2N-j)\times(j-1)}{2}+2(N-i)-1\right]\right)=\boldsymbol{A}(k,i,j) \tag{4}$$

右映射的计算方法如下：

将原图做一次映射，A' 表示映射后的图像：

$$A'(k,i,j)=\boldsymbol{A}(k,i,N-1-j) \tag{5}$$

通过左映射算法(1)~(4)，得到右映射算法。

把二维矩阵 $\boldsymbol{P}(k,m)$ 折叠成为三维矩阵 $B(k,i,j)$ 的折叠算法为：

$$B(k,i,j)=\boldsymbol{P}(k,i\times N+j) \tag{6}$$

3 图像加密和解密

将左映射和右映射的映射次数设计为密钥 Key，当取 Key 的第奇数位时，映射为左映射，且映射次数为 Key 的位数数字的值；同理，当取 Key 第偶数位时，映射为右映射，且映射次数为 Key 的位数数字的值。例如 $Key=53$ 表示先左映射 5 次，然后右映射 3 次。

3.1 图像加密

(1) 把二维图像读成三维二进制矩阵，设置好密钥 Key；

(2) 取出 Key 的一位数字，并根据数字位置、数字和式(1)~式(4)将三维图像 $\boldsymbol{A}(k,i,j)$ 转化成二维图像 $\boldsymbol{P}(k,m)$；

(3) 用图像置乱算法将二维图像 $\boldsymbol{P}(k,m)$ 置乱；

(4) 根据折叠法(式(6))把二维矩阵 $\boldsymbol{P}(k,m)$ 折叠成为三维矩阵 $\boldsymbol{B}(k,i,j)$，继续取出 Key 的下一位数；

(5) 重复(2)~(4)步，直到取完 Key 的位数，把最后生成的三维二进制矩阵读成二维十进制矩阵。

3.2 图像解密

(1) 从末位起读取 Key 的一位数字，并且把二维矩阵图像读成三维 $\boldsymbol{A}(k,i,j)$；

(2) 把三维矩阵 $\boldsymbol{A}(k,i,j)$ 拉伸成一个二维矩阵 $\boldsymbol{P}(k,m)$

$$\boldsymbol{P}(k,i\times N+j)=\boldsymbol{A}(k,i,j) \tag{7}$$

(3) 根据图像置乱算法的解密算法把 $\boldsymbol{P}(k,m)$ 解密，并且根据读取 Key 的第几位数值和数值的大小来判断加密是左映射还是右映射和解密次数。如果加密时是左映射，则解密方法如下：

当 $j\geqslant i$ 且 $N-j$ 是奇数时

$$\boldsymbol{A}(k,i,j)=\boldsymbol{P}\left(k,\left[\frac{(N+j+2)(N-j-1)}{2}+2(j-i)\right]\right) \tag{8}$$

当 $j \geqslant i$ 且 $N-j$ 是偶数时

$$\boldsymbol{A}(k,i,j)=\boldsymbol{P}(k,[\frac{(N+j+3)(N-j-2)}{2}+2(j-i)+1]) \quad (9)$$

当 $j<i$ 且 j 是偶数时

$$\boldsymbol{A}(k,i,j)=\boldsymbol{P}(k,[\frac{(N^2+N+(2N-j-1)\times j}{2}+2(N-i-1)]) \quad (10)$$

当 $j<i$ 且 j 是奇数时

$$\boldsymbol{A}(k,i,j)=\boldsymbol{P}(k,[\frac{(N^2+N+(2N-j)\times (j-1)}{2}+2(N-i)-1]) \quad (11)$$

若加密时是右映射，则解密方法如下：

首先利用左映射的解密方法进行解密，然后再根据式(5)的逆过程式(12)求出 $\boldsymbol{A}'(k,i,j)$：

$$\boldsymbol{A}'(k,i,N-1-j)=\boldsymbol{A}(k,i,j) \quad (12)$$

(4)继续读取 *Key* 的数值，重复(2)、(3)步，直到读取完 *Key* 的位数；

(5) 把最后生成的三维二进制矩阵转化为二维十进制矩阵，则其对应的图像为所求的解密图像。

4 加密实例和安全性能分析

4.1 图像加密

本文在加密过程中的第(3)步的置乱是基于改进的 Logistic 映射(其动力学方程为：$x_{n+1}=(\beta+1)(1+\frac{1}{\beta})^{\beta}\ x_n(1-x_n)^{\beta}$，式中：$\beta$ 为[1,4]之间的实数，$x_0 \in (0,1)$)。

对 $L=256$ 的 lena 灰度图进行加密，如图 4 所示，其直方图如图 5 所示。当 Logistic 映射中 $x_0=0.556\,465\,656$，$\beta=3.943\,534\,534\,534$ 时，*Key*=1 和 *Key*=01 时的加密效果分别如图 6 和图 7 所示，其对应的直方图分别如图 8 和图 9 所示。

由图 5、图 7、图 9 可以看出，本文提出的加密算法

图 4 原图

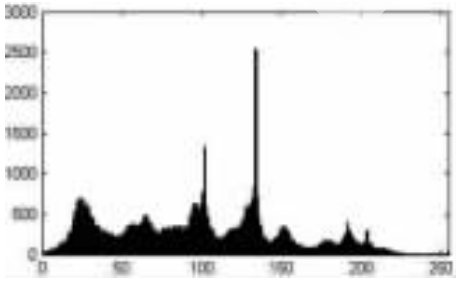

图 5 原图的直方图

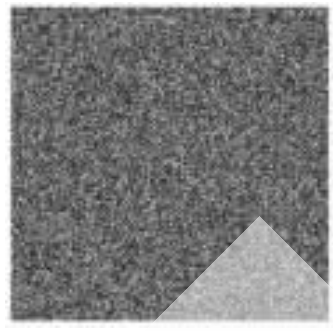

图 6 *Key*=1 密图

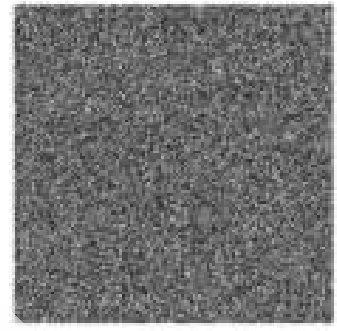

图 7 *Key*=01 密图

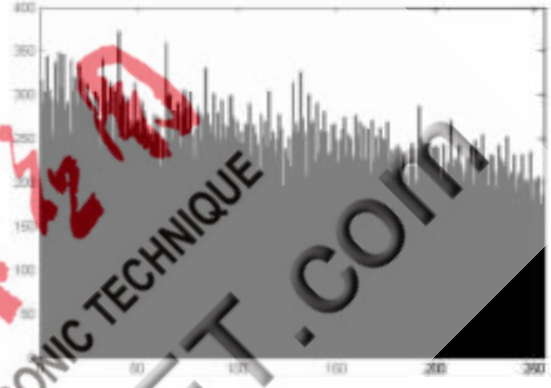

图 8 *Key*=1 直方图

加密的密图的直方图与原图的直方图是不同的，这说明这种加密算法改变了图像像素，隐藏了直方图信息，可以有效地抵御已知(选择)明文攻击，提高了加密算法的安全性。

4.2 安全性能分析

本算法中涉及了左右映射和改进的 Logistic 映射，其中左右映射的密钥空间只与密钥长度大小有关，它们之间的关系如表 1 所示。

表 1 密钥长度和密钥空间

密钥长度/位	64	128	256	512
密钥空间	1.84×10^{19}	3.4×10^{38}	1.16×10^{77}	1.34×10^{154}

(1) 密钥敏感性分析。用 *Key*=12 495 678 对图像进行加密，然后分别用 *Key*=12 495 677 和 *Key*=12 495 679 以及正确的 Logistic 映射密钥进行解密，得到的解密图像如图 10 和图 11 所示。即使加密密钥和解密密钥仅有非常小的差异，也无法解密图像，加密算法对密钥变化非常敏感。

(2) 统计分析。原始图像中相邻像素的相关性是很大的，为了破坏统计攻击，必须降低相邻像素的相关性，对图像中像素点相邻点进行相关分析。相关系数的计算方法见参考文献[7]。

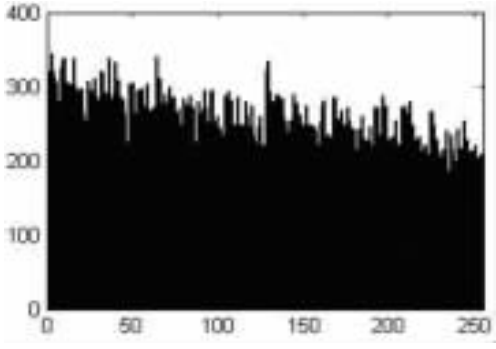

图 9 Key=01 直方图

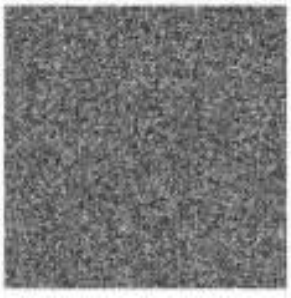

图 10 Key_1=12 495 677 解密图

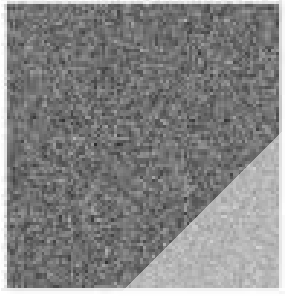

图 11 Key_2=12 495 679 解密图

经过计算,各方向相关系数如表 2 所示。显然,密图像素值与相邻点像素值之间的相关系数非常小,解密算法很难利用统计方法从密图中恢复原图。

表 2 加密前后,像素点(x,y)与其相邻点的相关系数

	原图	密图
对角线方向	0.925 5	0.005 4
水平方向	0.941 1	0.006 7
垂直方向	0.964 8	0.003 4

(3) 计算分析原图和密图的不动点比。原图像素点$A(i,j)$,在加密后其图像灰度值没有发生变化,则称该像素点为不动点。图像中不动点占有像素的百分比称为该图像的不动点比。在 Key=1 时,即图 6 和图 4 的不动点比为 0.45%;Key=01 时,即图 8 和图 4 的不动点比为 0.43%。这说明只在一次左映射或者右映射的情况下,不动点的数目很少,99.4%以上的像素点都发生了改变,置乱效果很明显。

本文提出了一种基于三维空间下的图像加密算法,算法具有如下优点:

(1) 公式非常简单,容易编程实现;

(2) 映射是可逆的;

(3) 在加密/解密过程中没有信息损失;

(4) 加密的密钥基本没有限制;

(5) 密图和原图大小一致,没有大小差异;

(6) 能满足实时需要,适合大尺寸图像加密;

(7) 经过映射后,可以改变原图像的直方图;

(8) 加密算法简单,容易硬件实现。

参考文献

[1] 钮心忻.信息隐藏与数字水印[M].北京:北京邮电大学出版社,2004.

[2] 孙鑫,易开祥,孙优贤.基于混沌系统的图像加密算法[J].计算机辅助设计与图形学学报,2002,14(2):136-139.

[3] 陶棕,李之棠.混沌加密图像算法[J].计算机工程与科学,2003,25(4):7-9.

[4] 李娟,冯勇,杨旭强.三维可逆混沌映射的图像加密算法.光学技术,34(6):918-923.

[5] 刘家胜.基于混沌的图像加密技术研究[D].安徽大学,2007:55-56.

(收稿日期:2009-12-15)

作者简介:

农盛功,男,1984 年生,硕士研究生,主要研究方向:图像安全。

周满元,男,1971 年生,教授,博士,主要研究方向:CAD/CAM,快速成型制造,企业信息化,信息安全。