计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2018,27(7):177−181 [doi: 10.15888/j.cnki.csa.006402]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

# 基于格雷码和混沌系统的图像加密算法①

薛　伟, 吕　群

(江南大学 物联网工程学院, 无锡 214122)
通讯作者: 吕　群, E-mail: wflv2015@163.com

**摘　要**: 针对当前的一些具有置乱-扩散结构的图像加密算法中, 存在加密程度低的问题, 提出了一种新的图像加密算法. 置乱阶段先利用格雷码变换进行全局的置乱, 然后再根据混沌序列进行行列间置乱变换. 扩散部分采用了正向和反向的异或操作. 实验结果表明, 该算法提高了密文的安全性和密钥的敏感性, 增大了密钥空间的大小, 同时能很好的抵抗统计分析、穷举攻击等.

**关键词**: 图像加密; 格雷码; 混沌系统; SHA-384; 安全性分析

引用格式: 薛伟,吕群.基于格雷码和混沌系统的图像加密算法.计算机系统应用,2018,27(7):177–181. http://www.c-s-a.org.cn/1003-3254/6402.html

## Image Encryption Algorithm Based on Gray Code and Chaotic System

XUE Wei, LYU Qun

(School of Internet of Things Engineering, Jiangnan University, Wuxi 214122, China)

**Abstract**: Aiming at solving the problem that some image encryption algorithms with the permutation-diffusion structure have low degree of encryption, this paper presents a new image encryption method. In the scrambling stage, firstly global scrambling uses gray code transformation, then scrambles pixel among row and column by chaos sequence. Diffusion part adopts the forward and reverse exclusive OR operation. The experimental results show that the proposed algorithm enhances the security of the cipher-image, key sensitivity, and increases the size of the key space, at the same time it can resist the statistical analysis, brute force attack.

**Key words**: image encryption; gray code; chaotic system; SHA-384; security analysis

## 1　引言

随着计算机技术和网络技术的发展, 信息安全的问题日益凸显[1]. 数字图像具有数据量大、空间有序、相关性强、冗余度高的特点[2], 使用传统的算法(DES、AES)时加密效率比较低. 混沌系统具有一些良好的特性, 使得其在图像加密领域越来越受欢迎. 当今有很多基于混沌系统的图像加密算法被提出.

Xu 等人[3]提出一种对图像分块置乱以及对像素点动态索引进行扩散的图像加密算法; Norouzi 等人[4]提出一种对行列分块置乱以及对位平面进行扩散的图像加密算法; Wang 等人[5]根据混沌序列和万有引力定律提出了一种图像加密算法; Xu 等人[6]根据混沌系统对经过位平面分解后的明文图像进行置乱与扩散. 这几种加密算法总体来说加密效果都不错, 但是混沌序列都与明文图像无关, 因此不能很好的抵抗选择明文攻击. 林等人[7]提出了一种基于位平面自适用参数的图像加密算法, 该算法产生混沌序列时引入了明文图像, 可以在一定程度上抵御选择明文攻击, 但是该算法只是简单的把图像灰度值的总和引入到混沌系统的初始值中, 算法的安全性还可以提高. 本文结合当今一些算法存在的安全性问题, 根据混沌系统、格雷码和 SHA-384 提出了一种基于置乱-扩散模式的图像加密算法.

① 基金项目: 国家自然科学基金 (61374047)
Foundation item: National Natural Science Foundation of China (61374047)
收稿时间: 2017-10-26; 采用时间: 2017-11-14; csa 在线出版时间: 2018-06-27

实验结果表明, 该算法具有较好安全性.

## 2 算法基础

### 2.1 $(n,k,p)$-格雷码

本文中使用了$(n,k,p)$-格雷码[8]. 假设两个非负整数$A$, $G$可以分别分解为以$n$为基数的$k$位序列, 如果整数$G$的序列满足下面的表达式, 那么整数$G$的序列就可以看作是整数$A$序列的$(n,k,p)$-格雷码.

$$\begin{cases} g_i = a_i, \text{ if } i > k-p-2 \\ g_i = \mod((a_i + a_{i+p+1}), n), \text{ if } 0 \leqslant i \leqslant k-p-2 \end{cases} \tag{1}$$

上式中$0 \leqslant i \leqslant k-1$, $n \geqslant 2$, $0 \leqslant p \leqslant k-2$, $p$称为间隔参数. $\mod(x,y)$表示$x$除以$y$得到的余数.

在下文中, 规定$(n,k,p)$-格雷码中的 $n$ 为 2, 因此可以把参数 $n$ 去掉, 变为$(k,p)$-格雷码. 同时规定用$\left(\bar{X},k,p\right)$表示把非负整数$\bar{X}$经过$(k,p)$-格雷码转换后得到的整数数值. 在文中设计了利用$(k,p)$-格雷码变换对图像进行全局置乱.

### 2.2 混沌系统

Lorenz 系统的表达式如 (2) 所示.

$$\begin{cases} \dot{x} = a(y-x) \\ \dot{y} = cx - y - xz \\ \dot{z} = xy - bz \end{cases} \tag{2}$$

其中$x$, $y$, $z$表示系统的状态, $a$, $b$, $c$表示的是系统参数. 当$a=10$, $b=8/3$, $c=28$时, 系统会进入混沌状态.

## 3 图像加密过程

### 3.1 密钥的产生

在本文的加密算法中, 通过明文灰度值和 SHA-384 产生了一个 384 位的密钥. 把这 384 位的密钥按每 8 位为一个整数 $(k_i)$ 进行划分, 密钥$K$可以表示为如下的形式: $K = k_1, k_2, k_3, \cdots, k_{48}$.

混沌系统的初始值计算如下:

$$x_0 = x_0' + \mod\left(\frac{(k_1 \oplus \ldots \oplus k_8) + \sum_{i=1}^{48} k_i/48}{256}, 1\right) \tag{3}$$

$$y_0 = y_0' + \mod\left(\frac{(k_9 \oplus \ldots \oplus k_{16}) + \sum_{i=1}^{48} k_i/48}{256}, 1\right) \tag{4}$$

$$z_0 = z_0' + \mod\left(\frac{(k_{17} \oplus \cdots \oplus k_{24}) + \sum_{i=1}^{48} k_i/48}{256}, 1\right) \tag{5}$$

其中$\oplus$表示异或操作, $x_0'$, $y_0'$, $z_0'$为给定值可以看作是密钥的一部分.

扩散部分的初始值计算如下:

$$u_0 = \mod\left(\frac{(k_{25} \oplus \ldots \oplus k_{32}) + \sum_{i=1}^{48} k_i/48}{256}, 1\right) \tag{6}$$

$$v_0 = \mod\left(\frac{(k_{33} \oplus \ldots \oplus k_{40}) + \sum_{i=1}^{48} k_i/48}{256}, 1\right) \tag{7}$$

### 3.2 像素位置置乱

#### 3.2.1 图像全局位置置乱

步骤 1. 假设明文图像和置乱图像长度都是 $M \times N$, 把图像转换成一维的数组, 选取数组长度的基数为 2, 会得到 $M \times N = 2^k$.

步骤 2. 令$\bar{X} = 0$.

步骤 3. 对于$\bar{X}$, 计算出相应的经过格雷码变换后的数值$\left(\bar{X},k,p_1\right)$以及$\left(\bar{X},k,p_2\right)$, 其中 $p_1$ 和 $p_2$ 的大小人为设定, 并且可以当作是密钥的一部分.

步骤 4. 在明文图像中找到第$\left(\bar{X},k,p_1\right) \oplus F_1$个像素点并把它转移到置乱图像中第$\left(\bar{X},k,p_2\right) \oplus F_2$个像素点所在的位置. $F_1$ 和 $F_2$ 可在$(0, 2^k - 1)$范围内取值.

步骤 5. 令$\bar{X} = \bar{X} + 1$然后重复步骤 3 和步骤 4, 直到$\bar{X} = 2^k - 1$时所有的像素点完成置乱, 并转换为 $M \times N$ 的二维矩阵 **B**.

#### 3.2.2 图像行列间位置置乱

步骤 1. 根据公式 (2)、公式 (3)、公式 (4) 和公式 (5) 产生长度为 $M \times N$ 的混沌序列 $X$, $Y$, $Z$. 将序列 $X$, $Y$ 转换为和矩阵 **B** 一样大小的二维矩阵 **X'**, **Y'**.

步骤 2. 对矩阵 **X'**逐行进行排序, 进而会得到排序以后的位置矩阵 **IR**, 然后用矩阵 **IR** 对图像 **B** 逐行进行行置乱, 最后得到行置乱图像 $\mathbf{B}_1$.

步骤 3. 对矩阵 **Y'**逐列进行排序, 进而会得到相应的位置矩阵 **IC**, 然后用矩阵 **IC** 对矩阵 $\mathbf{B}_1$ 逐列进行列

置乱, 最后得到列置乱矩阵 **G**.

### 3.3 像素扩散

步骤 1. 根据公式 (6)、公式 (7) 获得两个初始值 $u_0$ 和 $v_0$, 同时设置控制参数 $b$ 和 $c$.

步骤 2. 令 $i$=0.

步骤 3. 用下面的公式得到 2 个 8 位的整数值 $d_i$ 和 $e_i$.

$$d_i = floor(L \times u_i) \tag{8}$$

$$e_i = floor(L \times v_i) \tag{9}$$

其中, $0 \le i \le M \times N/2-1$, $L$ 表示色阶, 例如对 8 位灰度图像来说 $L$=256. $floor(x)$表示小于或等于 $x$ 的最大整数.

步骤 4. 通过公式 (10) 和公式 (11), 更改像素的灰度值:

$$H(2i+1) = \Phi(2i+1) \oplus \bmod((d_i + H(2i)), L) \tag{10}$$

$$H(2i+2) = \Phi(2i+2) \oplus \bmod((e_i + H(2i+1)), L) \tag{11}$$

其中, $\Phi(2i+1)$和$\Phi(2i+2)$是当前需要操作的像素值, $\Phi$是由矩阵 **G** 转换的一维数组, 大小为 $M \times N$. $H(i)$表示已经经过扩散处理后的像素灰度值. $H(0)$ 由公式 (12) 获得.

$$H(0) = floor((k_{41} + \cdots + k_{44})/4) \tag{12}$$

步骤 5. 通过改进的 Arnold 变换计算下一次的 $(u_{i+1}, v_{i+1})$, 如公式 (14) 所示. 改进的 Arnold 变换增加了一个参数 $t$, 而参数 $t$ 由上一个经过扩散处理后的灰度值得到, 如 (13) 所示. 由公式 (13)、(14) 得到的 $(u_{i+1}, v_{i+1})$与其他灰度值相关联能更好的体现出扩散效果.

$$t = \bmod(H(2i+1), 3) + 1 \tag{13}$$

$$\begin{pmatrix} u_{i+1} \\ v_{i+1} \end{pmatrix} = \bmod\left(\begin{pmatrix} 1 & b \\ c & 1+bc \end{pmatrix}^t \begin{pmatrix} u_i \\ v_i \end{pmatrix}, 1\right) \tag{14}$$

步骤 6. 令 $i$=$i$+1, 重复步骤 3–步骤 5, 直到所有的像素点的灰度值都完成更改.

步骤 7. 对 3.2.2 节步骤 1 中产生的混沌序列 $Z$ 作如下的处理, 得到反向扩散序列 $W(i)$:

$$W(i) = \bmod\left(\left(floor\left(Z(i) \times 10^{14}\right)\right), L\right) \tag{15}$$

其中, $1 \le i \le M \times N$.

步骤 8. 通过下面的公式反向改变像素的灰度值:

$$C(i) = C(i+1) \oplus \bmod((H(i) + W(i)), L) \tag{16}$$

其中, $i = M \times N, \cdots, 2, 1$. 把得到的 $C(i)$ 转化为 $M \times N$ 的二维矩阵, 也就是最终的加密图. $C(M \times N+1)$ 可以由公式 (17) 获得.

$$C(M \times N + 1) = floor((k_{45} + \cdots + k_{48})/4) \tag{17}$$

图 1 是整个加密过程的结构图. 解密过程与加密过程类似, 对密文图像实行相反的操作, 就可以恢复出明文图像.

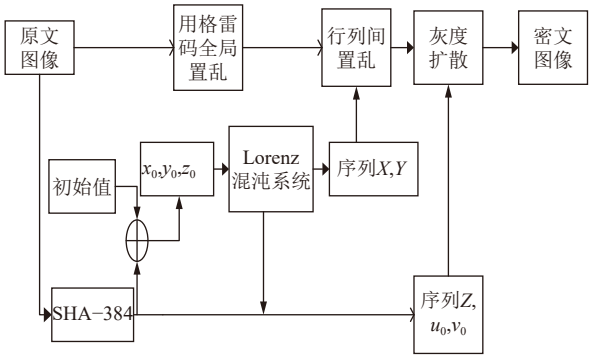
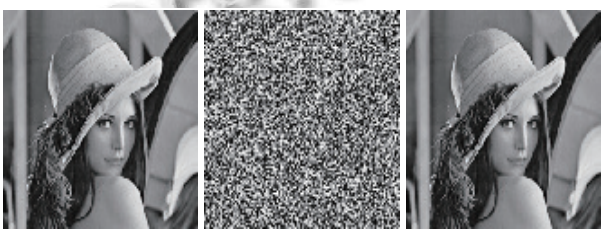



图 1 图像加密结构图

## 4 实验仿真与分析

在本文的仿真过程中, 选择了大小为 256×256 的 Lena 灰度图进行仿真. 在加密系统中设置参数 $x'_0 = 10.01$, $y'_0 = 6.21$, $z'_0 = 20.38$, $p_1$=1, $p_2$=10, $F_1$=16 735, $F_2$=9366. 图 2 是实验仿真图. 为了评价算法的整体性能, 下面对算法进行安全性分析.



(a) 原图 (b) 加密图 (c) 解密图像

图 2 实验仿真图

### 4.1 图像直方图

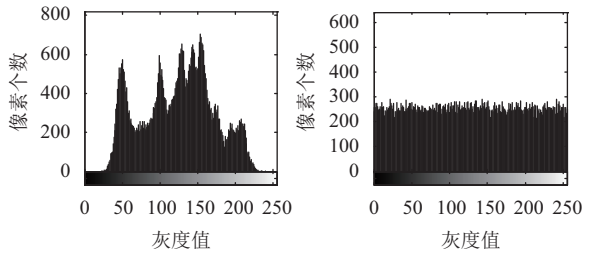
图像的直方图可以用来表示图像中所有像素点灰度值的分布状况. 图 3 是明密文图像的直方图.

从图 3(b) 中可以看出密文图像的直方图分布的比较均匀, 说明该算法可以很好的掩盖明文图像的灰度统计特性.

### 4.2 相邻像素相关性

相邻像素相关性表示的是图像中相邻像素间的相

关水平. 如果相关性越低, 那么抵抗统计攻击的能力越强. 为了检验图像中两个相邻像素点之间的相关性, 分别在 Lena 明密文图像的水平、垂直以及对角线方向上随机抽取 2000 对相邻的像素点, 并计算相关系数, 结果如表 1 所示.




(a) 明文图像直方图　　(b) 密文图像直方图

图 3　图像直方图分析

表 1　相邻像素相关系数及比较

| 方向 | 原图 | 加密图 | 文献[5] | 文献[6] | 文献[7] |
|---|---|---|---|---|---|
| 水平 | 0.9568 | 0.0021 | 0.0129 | –0.0230 | 0.0062 |
| 垂直 | 0.9662 | 0.0014 | 0.0065 | 0.0019 | 0.0041 |
| 对角 | 0.9177 | –0.0012 | 0.0013 | –0.0034 | –0.0032 |

表 1 中的数据, 数值越接近 1 表示相关性越高, 越接近 0 表示相关性越低. 通过比较, 本文的算法能有效的降低相邻像素点间的相关性.

### 4.3　密钥空间及敏感性

一个好的加密算法应该是有一个足够大的密钥空间, 以抵抗穷举攻击. 本文的密钥主要由 2 部分构成: 给定的初始值 $x'_0$, $y'_0$, $z'_0$; 384 位长的哈希值. 对于以上的 3 个初始参数, 如果数据精度为 $10^{-15}$, 那么密钥空间至少为 $S = 2^{384} \times 10^{45} \approx 3.9 \times 10^{160}$, 足可以抵抗穷举攻击.

为了验证敏感性, 使初始值 $x_0' = 10.01 + 10^{-15}$, 其他参数保持不变, 对密文图像进行解密. 图 4(c) 表示用错误的密钥的解密图, 与图 4(a) 的原图完全不像, 表示解密失败, 可见微小的差别也会导致解密失败, 从而说明该算法有较好的密钥敏感性.

(a) 原图　　(b) 加密图　　(c) 解密图

图 4　密钥敏感性测试

### 4.4　差分攻击

图像加密中一般使用 NPCR(像素变化率)、UACI(平均改变强度) 来评价算法抗差分攻击的性能. 对于一幅 256 级的灰度图像, NPCR 的值大于 99.6094%, UACI 的值大于 33.4635% 时算法才是安全的. 实验中随机选取了 5 个像素点, 其灰度值都改变 1, 加密轮数为 1 轮, 计算相应的 NPCR 和 UACI, 结果见表 2, 均值见表 3. 通过比较可以看出, 本文算法的 NPCR 和 UCAI 都能满足算法安全的要求, 从而可以较强的抵抗差分攻击.

表 2　不同位置的 NPCR 和 UACI(单位: %)

| 坐标 | 4,86 | 141,36 | 216,66 | 136,200 | 98,146 |
|---|---|---|---|---|---|
| NPCR | 99.66 | 99.60 | 99.61 | 99.65 | 99.65 |
| UACI | 33.49 | 33.61 | 33.55 | 33.45 | 33.58 |

表 3　NPCR 和 UACI 的均值及比较 (单位: %)

| | 本文算法 | 文献[5] | 文献[6] | 文献[7] |
|---|---|---|---|---|
| NPCR | 99.63 | 99.63 | 99.62 | 99.59 |
| UACI | 33.54 | 33.51 | 33.51 | 33.43 |

### 4.5　抗选择明文攻击分析

本文算法的中间密钥可以看作是矩阵 **IR** 和 **IC**(置乱部分的中间密钥)、序列 $W$(扩散部分的中间密钥), 而得到这 3 个中间密钥需要迭代混沌系统, 混沌系统的初始值与 SHA-384 和明文图像共同产生的散列值有关, 因此可以认为本文算法的中间密钥和原始图像有关. 选择不同的明文图像得出的中间密钥是不相同的, 因此以特定明文图像得出的中间密钥并不能破解其他的密文图像, 所以说本文算法可以较好的抵抗选择明文攻击.

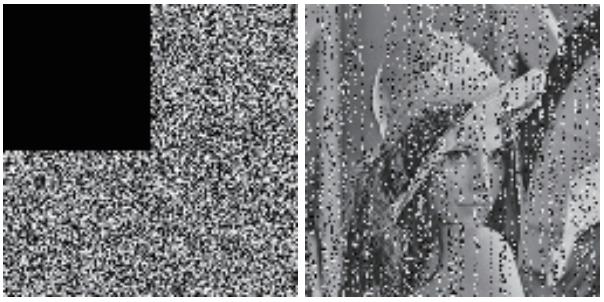
### 4.6　剪切攻击

在本文的算法中由于置乱过程的作用, 使得密文图像中被剪切掉的那部分相应的错误解密图像会均匀的分布在整个解密图像中, 因此密文图像即使受到剪切攻击后, 在解密图像中也可以看到明文图像的内容. 图 5(a) 为密文图像被剪切掉四分之一的面积, 图 5(b) 是相应的解密图像, 从图中可以看出, 即使密文图像有丢失, 解密图像中的内容大致也可以被识别, 因此可以认为本算法具有一定的抗剪切攻击的能力.

## 5　结论

本文提出的图像加密算法有以下三个特点: 首先,

使用 SHA-384 和明文图像产生加密过程的密钥, 使混沌序列与明文图像有关, 能更好的抵抗选择明文攻击, 同时增大了密钥空间; 其次, 算法在置乱阶段使用了基于格雷码和混沌序列相结合的置乱方法, 可以更好的实现置乱效果; 最后, 算法中加入了正反向的扩散操作, 使得密文图像中的像素点与前后像素点有关, 能更好的掩盖图像的灰度统计特性. 实验结果表明, 本文的算法具有较好的安全性, 在图像传输领域有一定的潜在应用价值.



(a) 剪切图像 (b) 解密图像

图 5 抗剪切测试

## 参考文献


1 毛骁骁, 孙克辉, 刘文浩. 基于分数阶统一混沌系统的图像加密算法. 传感器与微系统, 2017, 36(6): 138–141.

2 文昌辞, 王沁, 苗晓宁, 等. 数字图像加密综述. 计算机科学, 2012, 39(12): 6–9, 24. [doi: 10.3969/j.issn.1002-137X.2012.12.002]

3 Xu L, Gou X, Li Z, *et al*. A novel chaotic image encryption algorithm using block scrambling and dynamic index based diffusion. Optics and Lasers in Engineering, 2017, (91): 41–52. [doi: 10.1016/j.optlaseng.2016.10.012]

4 Norouzi B, Seyedzadeh SM, Mirzakuchaki S, *et al*. A novel image encryption based on row-column, masking and main diffusion processes with hyper chaos. Multimedia Tools and Applications, 2015, 74(3): 781–811. [doi: 10.1007/s11042-013-1699-y]

5 Wang XY, Wei N, Zhang DD. A novel image encryption algorithm based on chaotic system and improved gravity model. Optics Communications, 2015, (338): 209–217. [doi: 10.1016/j.optcom.2014.10.042]

6 Xu L, Li Z, Li J, *et al*. A novel bit-level image encryption algorithm based on chaotic maps. Optics and Lasers in Engineering, 2016, (78): 17–25. [doi: 10.1016/j.optlaseng.2015.09.007]

7 林振荣, 倪骅波, 谌志涛. 基于位平面的自适应参数图像加密算法. 计算机工程与设计, 2016, 37(3): 597–601, 642.

8 Zhou YC, Panetta K, Agaian S, *et al*. (n, k, p)-Gray code for image systems. IEEE Transactions on Cybernetics, 2013, 43(2): 515–529. [doi: 10.1109/TSMCB.2012.2210706]