文章编号:1001 - 9081(2005)03 - 0543 - 03

一种基于混合混沌动力系统的图像加密算法

彭　飞,丘水生,龙　敏
(华南理工大学 电子与信息学院,广东 广州 510640)
(eepengf@ yahoo. com. cn)

摘　要:讨论了一种基于 Lorenz 系统、Chen's 系统以及 Lü 系统的混合混沌动力系统的图像加密算法。该算法通过密钥映射产生混合混沌动力系统的初始条件,利用混合混沌动力系统产生的混沌信号对图像信号进行掩盖加密。仿真与讨论结果表明,该算法密钥空间大,混沌信号的产生极其敏感地依赖于密钥,具有较好的加密效果和加密效率,并对统计分析具有较好的安全性。

关键词:混沌;混沌动力系统;混沌加密;图像加密
中图分类号: TP309.7　**文献标识码**:A

Image encryption algorithm based on mixed chaotic dynamic systems

PENG Fei, QIU Shui-sheng, LONG Min
(*College of Electronics & Information, South China University of Technology, Guangzhou Guangdong* 510640, *China*)

Abstract: The paper proposed an image encryption algorithm based on mixed chaotic dynamic systems including Lorenz system, Chen system and Lü system. With the map of the external keys, the initial condition of the mixed dynamic systems was acquired and chaotic signals were generated subsequently, then the plain image was encrypted by masking with the chaotic signals. The simulation and analysis results show that the chaotic signals are greatly sensitive to the keys and the algorithm has a large space of keys to acquire good encryption effects and efficiency, and a better security performance to statistical analysis.

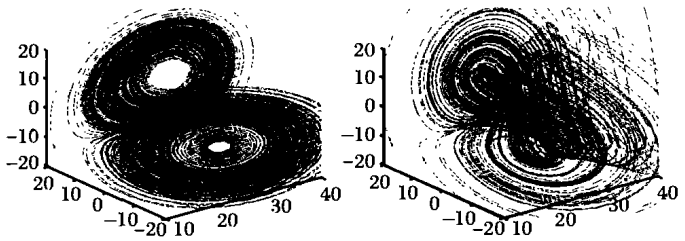
Key words: chaos; chaotic dynamic system; chaotic encryption; image encryption

0　引言

随着 Internet 技术与多媒体技术的飞速发展,多媒体通信已逐渐成为信息交流的重要手段。人们通过网络交流各种信息,进行网上贸易等。因此,信息的安全与保密显得越来越重要[1~3]。

混沌现象是在非线性动力系统中出现的确定性的、类似随机的过程,这种过程既非周期又不收敛,并且对初始值具有极其敏感的依赖性,因此采用混沌信号的信息加密技术具有广阔的应用前景。

混沌保密通信系统目前已有一定的进展[4],将混沌保密通信系统拓展至 Internet 及多媒体安全业务是当前混沌保密通信系统研究的一个方向[5]。



(a) Lorenz吸引子　(b) Chen 吸引子

图 1　三种混沌动力系统的吸引子

文献中已有许多采用混沌信号对图像进行加密的算法普遍采用单一混沌动力系统,且混沌加密的密钥空间设计问题尚未得到很好的解决。目前,提高混沌信号随机性是国际上的一个研究热点。本文讨论了一种基于 Lorenz 系统、Chen's 系统以及 Lü 系统的混合混沌动力系统的图像加密算法。该算法通过密钥映射产生混合混沌动力系统的初始条件,利用混合混沌动力系统产生的混沌信号对图像信号进行掩盖加密。仿真与讨论结果表明,该算法密钥空间大,混沌信号的产生极其敏感的依赖于密钥,具有较好的加密效果和加密效率,并对统计分析具有较好的安全性。

1　算法原理

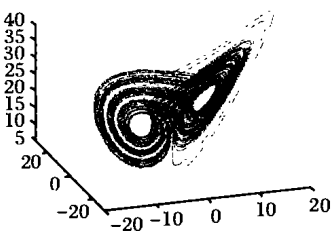
1.1　三个典型的混沌动力系统

Lorenz 系统、Chen 系统以及 Lü 系统均为三维自治系统,其动力学方程分别为:

$$\begin{cases}\dot{x} = a(y - x)\\ \dot{y} = cx - xz - y\\ \dot{z} = xy - bz\end{cases} \quad (1)$$

$$\begin{cases}\dot{x} = a(y - x)\\ \dot{y} = (c - a)x - xz + cy\\ \dot{z} = xy - bz\end{cases} \quad (2)$$

$$\begin{cases}\dot{x} = a(y - x)\\ \dot{y} = -xz + cy\\ \dot{z} = xy - bz\end{cases} \quad (3)$$



(c) L ü 吸引子

对于 Lorenz 系统,当参数取值为 $a = 10, b = 8/3, c = 28$ 时,系统有一个混沌吸引子[5],如图 1(a)所示;对于 Chen 系统,当参数取值为 $a = 35, b = 8, c = 28$ 时,系统有一个混沌吸

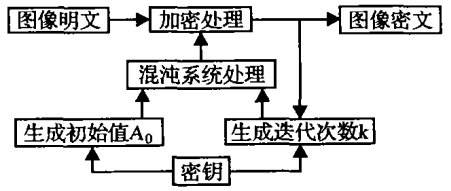
收稿日期:2004 - 08 - 12;**修订日期**:2004 - 11 - 05
基金项目:国家自然科学基金资助项目(60372004);广东省自然科学基金资助项目(31445,20820)
作者简介:彭飞(1977 -),男,湖南邵阳人,博士研究生,主要研究方向:混沌保密通信;　丘水生(1939 -),男,广东平远人,教授,博士生导师,主要研究方向:非线性系统理论、混沌保密通信、功率电子学;　龙敏(1977 -),女,湖南湘潭人,博士研究生,主要研究方向:保密通信、混沌理论及其应用.

引子[5]，如图 1(b)所示；对于 Lü 系统，当参数取值为 $a = 36, b = 3, c = 20$ 时，系统有一个混沌吸引子[5]，如图 1(c)所示。

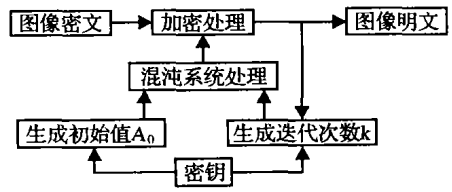
也就是说，由初始条件 (x_0, y_0, z_0) 在 Lorenz 系统、Chen 系统以及 Lü 系统通过四阶龙格－库塔法迭代作用下所产生的序列 $\{(x_k, y_k, z_k) \mid k = 0, 1, 2, 3, \cdots\}$ 是非周期、不收敛的，且对初始值非常敏感。

1.2 算法描述

由于混沌信号的产生主要依赖系统初值 (x_0, y_0, z_0)，以及迭代次数 k，因此目前很多基于混沌的图像加密算法均以上述参数作为加密系统的对称密钥，对图像的每一个像素点进行加密。针对这种情况，本文讨论的算法依靠外部密钥映射产生 Lorenz 系统、Chen 系统以及 Lü 系统初始值、迭代次数 k 和状态数 f。通过状态数 f 的值从三种混沌动力系统中选择其中一种进行 k 次迭代，得到 (x_k, y_k, z_k)，然后用经过处理后得到的三个混沌信号分别对像素的 R、G、B 三个值进行混沌加密处理。同时下一次系统的迭代次数依赖于前一次的密文和前一次迭代次数。这样加密图像的每一部分均依赖于外部密钥，从而提高了加密的可靠性。



(a) 加密过程



(b) 解密过程

图 2 图像的加密与解密过程

对于一幅 $M \times N$ 大小的图像，设获取每一像素的灰度值按照行、列的顺序组成明文 P，相应的密文用 C 可表示如下：

$$P = P_1 P_2 P_3 \cdots P_n \tag{4}$$

$$C = C_1 C_2 C_3 \cdots P_n \tag{5}$$

这里，$n = MN$。若第 i 个像素的明文和密文分别为 P_i 和 C_i，则 $P_i = (P_i^R, P_i^G, P_i^B)$。$P_i^R$，$P_i^G$，$P_i^B$ 分别代表明文 P_i 像素点的 R, G, B 值；$C_i = (C_i^R, C_i^G, C_i^B)$，$C_i^R$，$C_i^G$，$C_i^B$ 分别代表密文 C_i 像素点的 R, G, B 值。这里，$0 \leqslant P_i^R, P_i^G, P_i^B, C_i^R, C_i^G, C_i^B \leqslant 255$，$1 \leqslant i \leqslant n$。

设密钥为一 16 位的字符串，用 K 表示如下：

$$K = K_1 K_2 K_3 \cdots K_{16} \tag{6}$$

则加密过程：$C_i = (A_i - P_i) \bmod 256$ (7)

解密过程：$P_i = (A_i + C_i) \bmod 256$ (8)

这里，A_i 为经过 Lorenz 系统、Chen 系统以及 Lü 系统中某一混沌动力系统产生的一组混沌信号。其具体产生过程如下：

1) 在产生混沌信号前，首先通过下列公式产生一个处于[0,1]范围的实数 U, V, W 作为混沌系统的初始值：

$$T_1 = K_1 + K_2 + K_3 + \cdots + K_{16} \tag{9}$$

$$T_2 = ((K_1)_2 \oplus (K_2)_2 \oplus (K_3)_2 \oplus \cdots \oplus (K_{16})_2)_{10} \tag{10}$$

$$T_3 = T_1 * T_2 \tag{11}$$

$$U = (T_1 \bmod 256)/256 \tag{12}$$

$$V = (T_2 \bmod 256)/256 \tag{13}$$

$$W = (T_3 \bmod 256)/256 \tag{14}$$

$$k_1 = 5 + (\lfloor (T_1 + T_2 + T_3)/3 \rfloor \bmod 25) \tag{15}$$

$$f = T_3 \bmod 3 \tag{16}$$

这里，K_n，$(K_n)_2$，$(\cdot)_{10}$ 和 $\oplus$ 分别表示第 n 块密钥的 ASCII 值、第 n 块密钥的 ASCII 值的二进制表示、相应二进制表示的十进制值以及异或(Xor)处理；k_1 表示产生第一次混沌信号的迭代次数；$\lfloor \rfloor$ 表示向下取整；f 为标志位($f = 0, 1, 2$ 时分别选用 Lorenz 系统、Chen 系统和 Lü 系统产生混沌信号)。

2) 分别以 (U, V, W) 作为 Lorenz 系统、Chen 系统以及 Lü 系统的初始值，采用四阶龙格－库塔法迭代 T_1 次，分别得到 (x_{0L}, y_{0L}, z_{0L})、(x_{0C}, y_{0C}, z_{0C}) 和 (x_{0u}, y_{0u}, z_{0u})，以此作为 Lorenz 系统、Chen 系统以及 Lü 系统的初始值。

3) 根据 f 的值，选定相应的混沌动力系统，在初始值的基础上迭代 k_1 次，得到 (x_1, y_1, z_1)，同时以其作为下一次相应混沌动力系统迭代的初始值。

4) 通过下式可以得到 A_1，即第一组用于加密／解密的混沌信号，A'_1 为 A_1 转置：

$$A_1' = \begin{bmatrix} \lfloor (x_1 \bmod 1) * 10^4 \rfloor \\ \lfloor (y_1 \bmod 1) * 10^4 \rfloor \\ \lfloor (z_1 \bmod 1) * 10^4 \rfloor \end{bmatrix} \tag{17}$$

5) 以步骤 3)产生的 (x_1, y_1, z_1) 和密文为基础产生下一次迭代的次数 k_2 与标志位 f：

$$B_1' = \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} \lfloor (x_1 * 256 + C_1^R \rfloor \bmod 256 \\ \lfloor (y_1 * 256 + C_1^G \rfloor \bmod 256 \\ \lfloor (z_1 * 256 + C_1^B \rfloor \bmod 256 \end{bmatrix} \tag{18}$$

$$K_2 = 5 + (k_1 + b_1 + b_2 + b_3) \bmod 25 \tag{19}$$

$$f = ((b_1 + b_2 + b_3) * 5) \bmod 3 \tag{20}$$

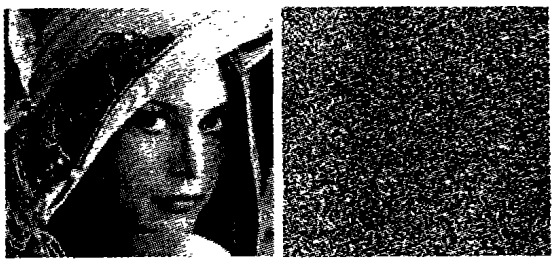
这里，B_1 为一中间值，B_1' 为其转置，b_1、b_2 与 b_3 为其元素。

6) 以 (x_1, y_2, z_3) 作为下一次迭代相应系统的初始值。

7) 重复 3)～5)，即可以产生所有的混沌信号。

2 算法仿真与分析

(a) 原始图像 (b) 加密图像



(c) 正确解密图像 (d) 错误解密图像

图 3 Lena 图像仿真结果

2.1 算法仿真

在对算法进行仿真时，选取 279 × 269 的 Lena 图像作为明文图像，如图 3(a)。加密/解密的密钥为长为 128 位的“ < * &^% &^% \;@ # $ 87?”，加密/解密结果如图 3(b)～(c)；若解密密钥作微小的变动，即更改为“ < * &^% &^% \;@ # $ 86?”，其解密结果如图 3(d)。整个仿真过程在 PIII1.3G，内

存为256M 的 Windows XP 系统中进行,采用 Matlab Version 6.5 Release 13 实现。

2.2 算法分析

2.2.1 密钥

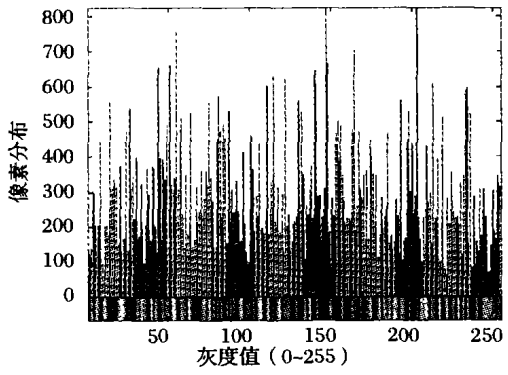
本文所讨论的图像加密算法,采用 128 位密钥,则密钥空间长度为: $2^{128} \approx 3.4 \times 10^{38}$,这得益于算法独特的结构即采用 128 位密钥映射混沌动力系统的初始值与迭代次数,因此本算法具有较强的抵抗穷举攻击的能力。和一些常见的混沌加密算法(一般采用单一混沌动力系统)相比较,本算法混合了三种混沌动力系统,在单一密钥下混沌系统及其迭代次数不断变化且混沌信号的产生极其敏感地依赖于密钥,因此相对于其他混沌加密算法,其置乱与分散程度更高。

算法对密钥的敏感度的试验结果如图 3(d),该图像是通过密钥"< * &%&%\;@#$ 86?"对图(b)进行解密的结果。显然,在密钥变化极小的情况下,解密结果与所期望的结果完全不同。

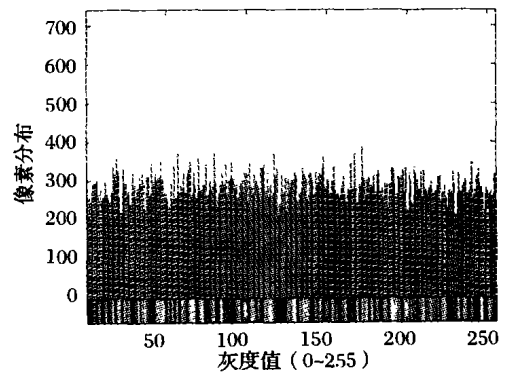
2.2.2 统计特性

Shannon 在文献[6]中指出"通过统计分析可能破译许多加密系统",并建议通过加强扩散与混淆抵抗基于统计分析的攻击。

通过对本文所介绍的图像加密算法进行统计分析,可以发现其扩散与混淆的特性对统计分析具有较强的抵抗力。这一点可以从原始图像与加密图像的直方图(如图 4(a) ~ (b))上可以看出,加密图像的直方图具有较好的均匀特性。



(a) 原始图像直方图



(b) 加密图像直方图

图4　原始图像及加密图像直方图比较

加密图像相邻像素的相关性分析。我们通过以下方法分别对加密图像的水平、垂直和对角方向相邻像素计算其相关性[8],具体按(21)、(22)式计算相关系数:

$$COV(x, y) = E(x - E(x))(y - E(y)) \tag{21}$$

$$R_{xy} = \frac{COV(x, y)}{\sqrt{Dx} \cdot \sqrt{Dy}} \tag{22}$$

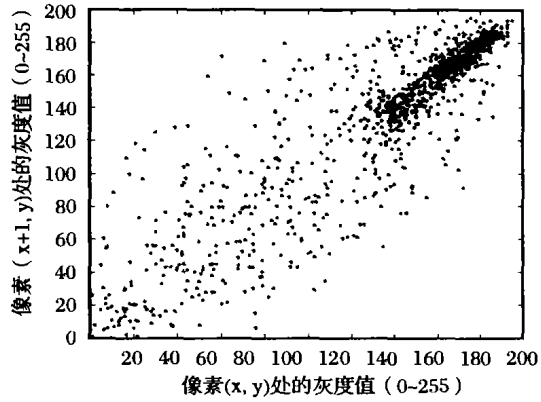
这里, $COV(x, y)$ 表示 x,y 的协方差; R_{xy} 表示 x,y 的相关系数。在实际的运算中,可以通过公式(23) ~ (25)进行:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{23}$$

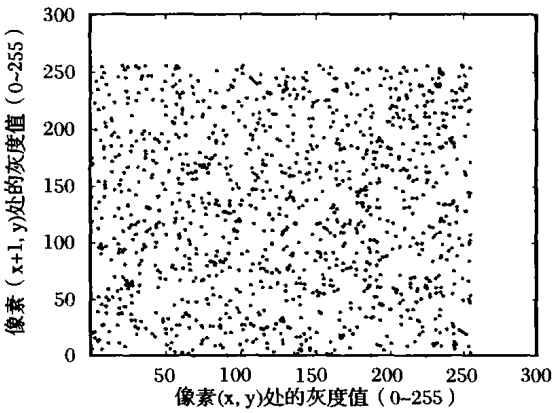
$$D(x) = \frac{1}{N}\sum_{i=1}^{n} (x_i - E(x))^2 \tag{24}$$

$$COV(x, y) = \frac{1}{N}\sum_{i=1}^{n} E(x_i - E(x))(y_i - E(y)) \tag{25}$$

式中 $E(x)$ 表示 x 的数学期望估计; $D(x)$ 表示 x 的方差估计; $COV(x, y)$ 表示 x,y 的协方差估计。



(a) 原始图像水平相关性



(b) 加密图像水平相关性

图5　原始图像和加密图像相邻像素点水平相关性

在原始图像与加密图像中随机选取 600 个像素点,计算其水平方向相邻两个像素之间的相关系数分别为 0.9847 和 0.0361,其分布如图 5 所示,同样的方法可以得出垂直方向和对角方向相邻两个像素之间的相关系数,见表 1。

表 1　相邻像素相关系数

像素关系	原始图像	加密图像
水平相邻	0.9847	0.0361
垂直相邻	0.9807	0.0022
对角相邻	0.9562	0.0324

由加密图像直方图的均匀特性和加密图像相邻像素的相关系数情况,可见所讨论的算法对加密信号具有较好的扩散与混淆能力,因此该算法能抵抗统计分析的攻击。

2.2.3 抗差分攻击能力的估计

通常,攻击者可能通过对图像作微小修改来观察解密结果的变化,这样可能发现原始图像和加密图像之间的某些关系。但是,如果对原始图像进行微小的改变会导致加密图像很大的变化,那么这种攻击方法就会变得非常无力。

(下转第 556 页)

```
    Public void Login() {};
    Public void Sign() {};
    Public string getSubject() {};
}                                                    //身份认证
public class DSA {
    public void Encryption(){};
    public void Decryption(){};
    public void Signature(){};
}                                                    //数字签名
```

1）服务实体 PeerA 在启动服务方法 StartApp( )后完成以下步骤：

a）首先实体 PeeA 和 PeeB 进行相互的身份认证，我们用 LoginContext 类中的方法完成身份认证，确认得到信任后，用 AdvertisementFactory 静态类来创建管道广告，通过创建一个新的 PipeID 实例来创建 ID，每个 PipeID 都包含它从中被创建的对等组的 ID。

```
myPipeAdv = ( PipeAdvertisement ) AdvertisementFactory.
    newAdvertisement
    ( PipeAdvertisement. getAdvertisementType( ) );
myPipeAdv. setPipeID( new PipeID( group. getID( ) ) );
if ( myPipeAdv == null) {
  println( "waitptext: Cannot create a Pipe Advertisement");
  return ShellApp. appMiscError;
}
println( "created a pipe advertisement...");
```

b）在创建了管道广告之后，我们必须创建一条基于该广告的输入管道，通过使用 jxta. pipe. Pipe 接口完成这项任务：

```
myInpPipe = pipes. createInputPipe ( myPipeAdv)
```

c）用原始的 JXTA 发现协议来发布这广告：

```
myDiscovery = group. getDiscovery( );
myDiscovery. publish( myPipeAdv, Discovery. ADV);
println( "published the pipe advertisement to the group...");
```

d）继续等待着接收输入消息，假定可以等到为止：

```
println( "waiting at the input pipe for a message...");
myMesg = myInpPipe. poll(0);
```

如果等待到了消息，用 DSA. Decryption( )方法解密消息，验证是否成功。

2）客户实体 PeerB 通过以下步骤发送消息

a）通过 getRemoteAdvertisements 方法查找远程服务 Servicename 的服务广告，将它放在本地缓存中：

```
myDiscovery. getRemoteAdvertisements
  (null, DiscoveryService. ADV, "name", Servicename, 1);
```

b）一段时间后，对等点开始从本地机器检索服务广告：

```
res = myDiscovery. getLocalAdvertisements
  ( DiscoveryService. ADV, "name"Servicename);
```

c）创建一条新消息：

```
Message dwMsg = pipes. createMessage( );
```

d）利用 DSA. Encryption ( )和 DSA. Signature( )方法对消息进行加密和数字签名，最后生成一个输出管道，通过输出管道发送消息：

```
dwoutpipe = pipes. createOutputPipe( myPipeAdv, )
dwoutpipe. sendMessage( dwMsg);
```

两个对等实体的通信结束，停止服务。

参考文献：

[1] DAVIDSON A. Peer-to-Peer File Sharing Privacy and Security [DB/OL]. Center for Democracy and technology, 2003.

[2] YEAGER B . Enterprise strength security on a JXTA P 2 P network [A]. Peer-to-Peer Computing 2003 (P2P 2003) Proceedings[C], 2003.7 - 8.

[3] 周功业，黎书生. 新一代网络计算模型——P2P 及其 JXTA 体系结构的设计与实现[J]. 计算机应用研究，2003,(9):139 - 142.

[4] 陈宇，唐旭章. 基于 P2P 系统的 JXTA 技术探析[J]. 计算机工程，2002,28(10):18 - 19.

[5] 邵丽炯，贺亮. 利用 JXTA 平台保障 P2P 安全的研究[J]. 微型电脑应用，2004,20(1):19 - 22.

(上接第 545 页)

一般采用两个参数对这种变化进行描述，即像素变化率(NPCR)和平均变化强度(UACI)[7]。对于一幅原始图像，其加密图像为 C_1，若对其修改一个像素点的灰度值进行加密的结果为 C_2，比较灰度值矩阵 C_1 和 C_2 所有点的值。如果 $C_1(i,j) = C_2(i,j)$，则 $D(i,j) = 1$，否则为 0。

则 *NPCR* 和 *UCAI* 可分别定义为：

$$NPCR = \frac{\sum_{i,j} D(i,j)}{M \times N} \times 100\% \quad (26)$$

$$UCAI = \frac{1}{M \times N} \sum_{i,j} \frac{| C_1(i,j) - C_2(i,j) |}{255} \times 100\% \quad (27)$$

其中，M 和 N 分别代表图像的长与宽。

以 Lena 图像的加密结果为 C_1，以经过修改后(改变第一个像素的灰度值)的 Lena 图像的加密结果为 C_2，通过计算得到其 $NPCR = 0.0051$，$UCAI = 0.2224$。这一结果表明对原始图像进行微小的改变会导致加密图像很大的变化，故算法具有较强的抵抗差分攻击能力。

正如文献[7]所指出，一个好的加密系统应该具有如下特征：对明文敏感、对密钥敏感以及将明文映射为随机暗文。这一点和混沌系统所具有的遍历性、对初值敏感性、对参数敏感性是很相似的。以上的讨论与分析表明，本文所讨论的算法具有较强的安全性和抵抗统计与差分攻击的能力。

2.2.4 算法效率

从算法上来看，本文所采用的加密/解密规则都具有迭代结构，适合于用计算机快速计算。从明文角度看，对于图像矩阵，也不需要先进行拉直预处理，省掉了预处理时间。从密钥的产生来看，本文算法采用 128 位自定义密钥通过映射产生混沌序列的初始条件，具有对选择密钥敏感的特性。又由于算法中所采用的混沌系统具有强耦合特性，因此，算法的迭代轮数不需要选择太多，这些都有效地提高了算法的执行效率。

参考文献：

[1] HOWARD C, LI XB. Partial encryption of compressed images and videos [J]. IEEE Transactions on Signal Processing, 2000, 48(8): 2439 - 2551.

[2] DANG PP, CHAN PM. Image encryption for secure Internet multimedia applications [J]. IEEE Transactions on Consumer Electronics, 2000, 46 (3) : 395 - 403.

[3] CHENG YJ, GUO JN. A new chaotic key based design for image encryption and decryption [A]. ISCAS 2000[C]. Geneva, Switzer-Land, 2000.

[4] JESSA M. Data encryption algorithms using one - dimensional chaotic maps[A]. ISCAS2000[C], 2000.28 - 31.

[5] 吕金虎，陆君安，陈士华. 混沌时间序列分析及其应用[M]. 武汉：武汉大学出版社，2002.224226.

[6] SHANNON CE. Communication theory of secrecy system[J]. Bell System Technical Journal, 1949, 28:656 - 715.

[7] CHEN GY, MAO YB, CHUI CK. A symmetric image encryption scheme based on 3D chaotic cat maps[J]. Chaos Solitons & Fractals, 2004,21:749 - 761.