文章编号：2095-2163(2023)05-0090-08 中图分类号：TP751 文献标志码：A

DNA 运算与超混沌系统新型图像加密算法

王梦生，孙先赫，马宏斌
（黑龙江大学 电子工程学院，哈尔滨 150080）

摘 要：由于传统图像加密的安全性有一定的局限，本文提出了一种以 DNA 计算、超混沌系统和哈希函数为基础混合模型的新型图像加密算法。该方法由 DNA 水平排列和扩散组成。DNA 水平排列基于 Logistic map 的映射函数应用于 DNA 图像，来改变 DNA 图像中元素所处的位置；而 DNA 水平扩散则定义两个新的代数 DNA 计算规则，称为 DNA 循环移位，使用了多种 DNA 计算规则，将置换后的 DNA 图像与密钥 DNA 图像进行扩散处理，再进行图像加密。实验表明，所提出的图像加密方案具有良好的加密效果。

关键词：图像加密；超混沌系统；哈希函数；DNA 计算；DNA 循环移位

DNA operation and new image encryption algorithm for hyper-chaotic system

WANG Mengsheng, SUN Xianhe, MA Hongbin
(School of Electronic Engineering, Heilongjiang University, Harbin 150080, China)

【Abstract】 The security of traditional image encryption is limited, so a new image encryption algorithm based on DNA computing, hyper-chaos system and hash function is proposed. The proposed scheme consists of horizontal arrangement and diffusion of DNA.DNA horizontal arrangement is based on the Logistic-map mapping function applied to the DNA image to change the position of elements in the DNA image.For horizontal DNA diffusion, two new algebraic DNA calculation rules are defined, which are called DNA cyclic displacement. We use a variety of DNA calculation rules. Firstly, the displaced DNA image is diffused with the key DNA image, and then the image is encrypted. Experimental analysis shows that the proposed image encryption scheme has good encryption effect.

【Key words】 image encryption; Chaotic system; Hash function; DNA calculation; DNA cyclic shift

0 引 言

近年来，基于 DNA 的图像加密由于具有大量并行性、海量存储等优点，引起了人们的关注[1]。由于仅使用基于 DNA 的图像加密并不安全，因此将 DNA 计算和混沌系统结合起来，以实现更安全的图像加密方法。但是，基于 DNA 计算和混沌系统的图像加密存在低维混沌中安全风险大、DNA 运算复杂程度低和加密速度慢等缺点[2]。本文提出了一种以 DNA 计算为基础，利用混沌系统和散列函数的混合模型作为新的图像加密方案。

1 基本原理

1.1 混沌系统

一维混沌系统有敏感性强的特性，被广泛运用在图像加密中。Logistic map 就是一维混沌系统的一种[3]，如式(1)：

$$x_{i+1} = a\,x_i(1 - x_i), x_i \in (0,1) \quad (1)$$

其中，x_i 表示当迭代时间为 i 时，x 的值。

当 $a \in (3.89,4]$ 时，x_i 始终位于 $[0,1]$ 内。此时，logistic map 适合选择 DNA 操作和 DNA 规则，并在更短的时间内执行置换过程。

超混沌系统与普通混沌系统不同，有两个或多个正 Lyapunov 指数[4]。此外，超混沌系统具有很强

基金项目：黑龙江省自然科学基金(YQ2020F012)。

作者简介：王梦生(1997-)，男，硕士研究生，主要研究方向：图像处理；孙先赫(1999-)，男，硕士研究生，主要研究方向：网络安全；马宏斌(1970-)，男，博士，副教授，主要研究方向：网络空间安全。
通讯作者：马宏斌 Email：mahongbin@hlju.edu.cn

收稿日期：2022-05-26

的机密性、较大的密钥空间和更复杂且不可预测的非线性行为，可帮助构建关键图像。本文使用公式(2)定义的超混沌系统：

$$\begin{cases} y_{i+1} = c_1 y_i + c_1 z_i \\ z_{i+1} = c_2 y_i + c_2 z_i + w_i - y_i u_i v_i \\ u_{i+1} = -c_3 z_i - c_4 u_i - c_5 v_i + y_i z_i v_i \\ v_{i+1} = -c_6 v_i + y_i z_i u_i \\ w_{i+1} = -c_7 y_i - c_7 z_i \end{cases} \tag{2}$$

其中，y_i, z_i, w_i, u_i, v_i，表示的是当迭代时间为 i 时，y, z, w, u, v 的值。

当系统参数 $C_i = [30, 10, 15.7, 5, 2.5, 4.45, 38.5]$ 时，系统呈现超混沌行为[5]。

1.2　脱氧核糖核酸序列

在生物学中，脱氧核糖核酸(DNA)是大部分生物的遗传物质[6]。在密码学中，也有着不可或缺的作用。DNA 主要由 4 种核酸组成，具体表示为腺嘌呤(A)、胞嘧啶(C)、鸟嘌呤(G)和胸腺嘧啶(T)，其中 A 和 T 互补，C 和 G 互补[7]。在计算机中，信息都是通过二进制存储，0 和 1 互补。因此，可以推断出 00 和 11 也有类似性质，10 和 01 也是如此。假如使用DNA中4个碱基对00、01、10和11进行编码，总共有8种规则可以相互配对，编码表见表1[8]。在计算机中，四进制是以 4 为基数的数字系统。将四位数字 0、1、2、3 与 A、C、G、T 之间一对一映射。

加密图像像素的灰度值可以用一个 4 位数的四进制数表示，使用表 1 的规则将其编码为长度为 4 的 DNA 序列。例如，十进制的 180 灰度值可以用四进制数“2310”来表示，由于数字 0、1、2、3 与 A、C、G 和 T 一一映射，最终转换成 $GTCA$。

表 1　DNA 编码表

Tab. 1　DNA code table

四进制	二进制	规则 1	规则 2	规则 3	规则 4	规则 5	规则 6	规则 7	规则 8
0	00	A	A	T	T	C	C	G	G
1	01	C	G	C	G	A	T	A	T
2	10	G	C	G	C	T	A	T	A
3	11	T	T	A	A	G	G	C	C

1.3　使用 DNA 计算的密码学操作

整个密码系统是由一个五维组 (P, C, K, Enc, Dec) 组成，其中明文空间用 P 表示，密文空间用 C 表示，K 是密钥空间，Enc 代表加密函数，Dec 则代表解密函数。加密函数将明文转换为密文，解密函数则相反。当 $C=0, T=1, A=2, G=3$ 时，各种不同的运算法则见表 2。

表 2　DNA 运算表

Tab. 2　DNA operation table

加	减	乘	异或	同或	右移	左移
加 $A\ C\ G\ T$	减 $A\ C\ G\ T$	乘 $A\ C\ G\ T$	异或 $A\ C\ G\ T$	同或 $A\ C\ G\ T$	右移 $A\ C\ G\ T$	左移 $A\ C\ G\ T$
$A\ C\ A\ T\ G$	$A\ C\ G\ T\ A$	$A\ T\ G\ C\ A$	$A\ A\ C\ G\ T$	$A\ T\ G\ C\ A$	$A\ A\ C\ G\ T$	$A\ A\ C\ G\ T$
$C\ A\ C\ G\ T$	$C\ A\ C\ G\ T$	$C\ G\ T\ A\ C$	$C\ C\ A\ T\ G$	$C\ G\ T\ A\ C$	$C\ T\ A\ C\ G$	$C\ C\ G\ T\ A$
$G\ T\ G\ C\ A$	$G\ T\ A\ C\ G$	$G\ C\ A\ T\ G$	$G\ G\ T\ A\ C$	$G\ C\ A\ T\ G$	$G\ G\ T\ A\ C$	$G\ G\ T\ A\ C$
$T\ G\ T\ A\ C$	$T\ G\ T\ A\ C$	$T\ A\ C\ G\ T$	$T\ T\ G\ C\ A$	$T\ A\ C\ G\ T$	$T\ C\ G\ T\ A$	$T\ T\ A\ C\ G$

在数学中，右循环移位是将一组数据重新排列，具体操作为将最后的数字移动到第一个位置，同时将所有其他条目移动到正确的位置。$R_{cs}(t, <c_0, c_1, \cdots, c_{n-1}>)$ 表示 t 次右循环移位，式(3)：

$$R_{cs}(t, \langle c_0, c_1, \cdots, c_{n-1}\rangle) = \langle c_{\mathrm{mod}(0-t,n)}, \cdots, c_{\mathrm{mod}(n-1-t,n)}\rangle \tag{3}$$

类似地，左循环移位是将第一个数据移动到最后一个数据的位置，同时将所有其他条目移到左侧位置。$L_{cs}(t, <c_0, c_1, \cdots, c_{n-1}>)$ 表示 t 次左循环移位，式(4)：

$$L_{cs}(t, \langle c_0, c_1, \cdots, c_{n-1}\rangle) = \langle c_{\mathrm{mod}(0+1,n)}, \cdots, c_{\mathrm{mod}(n-1+t,n)}\rangle \tag{4}$$

根据表 2 中 DNA 右移与左移运算法则，重新定义了两个基于 DNA 序列的新代数算子，称作 DNA 左移位和 DNA 右移位。根据表 1DNA 编码规则，定义 8 种 DNA 左右移位。如果考虑规则 1，则 $\langle c_0, c_1, c_2, c_3\rangle = \langle A, C, G, T\rangle$。

此外，假设 t 等于二进制代码 A、C、G 和 T 的十

进制值。

DNA 右循环移位 $f_{\rightarrow}$ 定义如式(5):

$$f_{\rightarrow}=\begin{bmatrix} R_{cs}((A)_{10},\langle A,C,G,T\rangle) \\ R_{cs}((C)_{10},\langle A,C,G,T\rangle) \\ R_{cs}((G)_{10},\langle A,C,G,T\rangle) \\ R_{cs}((T)_{10},\langle A,C,G,T\rangle) \end{bmatrix}=\begin{bmatrix} A & C & G & T \\ T & A & C & G \\ G & T & A & C \\ C & G & T & A \end{bmatrix} \tag{5}$$

DNA 左循环移位 $f_{\leftarrow}$ 定义如式(6):

$$f_{\leftarrow}=\begin{bmatrix} L_{cs}((A)_{10},\langle A,C,G,T\rangle) \\ L_{cs}((C)_{10},\langle A,C,G,T\rangle) \\ L_{cs}((G)_{10},\langle A,C,G,T\rangle) \\ L_{cs}((T)_{10},\langle A,C,G,T\rangle) \end{bmatrix}=\begin{bmatrix} A & C & G & T \\ C & G & T & A \\ G & T & A & C \\ T & A & C & G \end{bmatrix} \tag{6}$$

在编码过程中,如果在加密过程中使用 DNA 右循环移位,那么在解密过程则相反。

2 加密解密方案

加密方案的总体步骤如下:

步骤 1 通过原图像的哈希值和外部密钥 K 计算得出密钥 K' 与异或值 k_{xor};

步骤 2 将明文图像 $P1$ 编码成四进制图像 $P2$;

步骤 3 将四进制图像 $P2$ 编码为 DNA 图像 $P3$;

步骤 4 对 DNA 图像 $P3$ 进行 DNA 水平置换,得到置换后的 DNA 图像 $P4$;

步骤 5 计算密钥 DNA 图像 KDNA;

步骤 6 在置换 DNA 图像 $P4$ 和关键 DNA 图像 KDNA 之间进行 DNA 水平扩散,得到扩散后 DNA 图像 $P5$;

步骤 7 将扩散后 DNA 图像 $P5$ 解码为密码 DNA 图像 $P6$;

步骤 8 将密码 DNA 图像 $P6$ 解码为密码四进制图像 $P7$;

步骤 9 将密码四进制图像 $P7$ 解码为整数范围[0, 255]的密码图像 $P8$。

2.1 生成密钥以及初始值和密钥 DNA 图像

密码散列函数在图像加密系统领域发挥着基础性作用。安全散列算法(SHA-256)和消息摘要算法(MD5)是常见加密散列算法,256 位和 128 位散列值分别由上述算法产生。本文使用 SHA-256 和 MD5 组合成哈希函数来提高所提出加密方案的安全性。

在所提出的密码系统中,利用 256 位外部密钥和输入图像的哈希值来求出本文混沌系统的控制参数和初始值。将 256 位外部密钥 K 划分为十进制格式的 8 位块,如式(7):

$$K=\{k_1,k_2,\cdots,k_{32}\} \tag{7}$$

假设视图像大小为 $m\times n$ 的 2 维矩阵 $\boldsymbol{P}$,其中 $P(i,j)$ 表示位置(i,j) 处的像素值。计算3个大小为 m 的向量 $\boldsymbol{S}_1$、大小为 n 的 $\boldsymbol{S}_2$ 和大小为 $m+n-1$ 的 $\boldsymbol{S}_3$,$\boldsymbol{S}_1(i)$ 是 $\boldsymbol{P}$ 的第 i 行的所有像素值的总和,$\boldsymbol{S}_2(i)$ 是 $\boldsymbol{P}$ 的第 i 列所有像素值的总和,$\boldsymbol{S}_3(i)$ 表示的是 $\boldsymbol{P}$ 的第 i 对角线上的所有像素值的总和;将上述3个向量和外部密钥 K 哈希组合,生成 256 位哈希值 H,以十进制格式划分为 8 位块,如式(8):

$$\begin{cases} H=SHA256(MD5(\boldsymbol{S}_1),MD5(\boldsymbol{S}_2),MD5(\boldsymbol{S}_3),MD5(K)) \\ H=\{h_1,h_2,\cdots,h_{32}\} \end{cases} \tag{8}$$

之后,使用异或(XOR)操作将外部密钥 K 和哈希值 H 结合起来,生成新的密钥值 K' 和值 k_{xor},本文 K_{xor} 的值用 k_{xor} 来表示,如式(9)、式(10):

$$\begin{cases} K'=\{k_1',k_2',\cdots,k_{32}'\} \\ k_i'=h_i\oplus k_i \end{cases} \tag{9}$$

$$K_{xor}=k_1'\oplus k_2'\oplus\cdots\oplus k_{32}' \tag{10}$$

如果仅更改了普通图像或外部密钥的一位,则哈希值将完全不同。此外,图像结合密钥 K 和 SHA-256哈希值 H 后,暴力攻击需要 2^{256} 次才能破解。从 5 维超混沌系统中生成大小为 1 × 4 mn 的密钥 DNA 图像 KDNA,随后将其进行扩散处理。生成 KDNA 的步骤如下:

步骤 1 考虑 C_i = [30, 10, 15.7, 5, 2.5, 4.45, 38.5] 作为 5 维超混沌系统的控制参数;

步骤 2 依靠 k_{xor} 和密钥 K',计算 5维超混沌系统的初始值 $x(0)$,$y(0)$,z (0),$u(0)$,$w(0)$,如式(11);

$$\begin{cases} x(0)=(k_1'\oplus k_2'\oplus k_3'\oplus k_4'\oplus k_5'\oplus k_6'\oplus k_{xor})/256 \\ \mathrm{y}(0)=(k_7'\oplus k_8'\oplus k_9'\oplus k_{10}'\oplus k_{11}'\oplus k_{12}'\oplus k_{xor})/256 \\ z(0)=(k_{13}'\oplus k_{14}'\oplus k_{15}'\oplus k_{16}'\oplus k_{17}'\oplus k_{18}'\oplus k_{xor})/256 \\ u(0)=(k_{19}'\oplus k_{20}'\oplus k_{21}'\oplus k_{22}'\oplus k_{23}'\oplus k_{24}'\oplus k_{xor})/256 \\ w(0)=(k_{25}'\oplus k_{26}'\oplus k_{27}'\oplus k_{28}'\oplus k_{29}'\oplus k_{30}'\oplus k_{xor})/256 \end{cases} \tag{11}$$

步骤 3 预迭代 $(k_{31}'+k_{32}'+k_{xor})$ 次 5 维超混沌系统,以消除瞬态效应并增加安全性;

步骤 4 预迭代后,对 5 维超混沌系统迭代 $4\times(m\times n/5)$ 次,选取 4 mn 个首元素,得到大小为 1 × 4 mn 的伪随机向量 $\boldsymbol{PV}$;

步骤 5 通过式(12)将伪随机矩阵 $\boldsymbol{PV}$ 直接转换为大小为 1×4 mn 的密钥 DNA 图像 KDNA。

$$K_{DNA}=\begin{cases}A & 0.00 \leqslant \boldsymbol{PV} \leqslant 0.25\\ T & 0.25 \leqslant \boldsymbol{PV} \leqslant 0.50\\ C & 0.50 \leqslant \boldsymbol{PV} \leqslant 0.75\\ G & 0.75 \leqslant \boldsymbol{PV} \leqslant 1.00\end{cases} \tag{12}$$

2.2　原始图像的 DNA 编码

设原始图像 $\boldsymbol{P}1$ 是一个 2 维的大小为 $m \times n$ 矩阵。首先将普通图像 $\boldsymbol{P}1$ 转换为大小为 1×4 mn 的四维矩阵 $\boldsymbol{P}2$；其次，四维矩阵 $\boldsymbol{P}2$ 编码大小为 1×4 mn 的 DNA 图像 $\boldsymbol{P}3$。编码步骤为：

步骤 1　输入大小为 $m \times n$ 的普通图像 $\boldsymbol{P}1$；

步骤 2　将大小为 $m \times n$ 的平面图像 $\boldsymbol{P}1$ 重塑为大小为 $1 \times mn$ 的向量 $\boldsymbol{P}1$；

步骤 3　通过将 $\boldsymbol{P}1$ 每个像素值编码为 4 位四进制数，将向量 $\boldsymbol{P}1$ 转换为大小为 1×4 mn 的四进制矩阵 $\boldsymbol{P}2$；

步骤 4　根据 k_{xor} 值和密钥 K' 计算 logistic map 的控制参数 u 和初始值 $x(0)$，式(13)：

$$\begin{cases}u = 3.89 + (k_1' \oplus k_2' \oplus k_3' \oplus k_4' \oplus k_5' \oplus k_6' \oplus \\ \qquad k_7' \oplus k_8' \oplus k_{xor})/256 \times 0.01\\ x(0) = (k_9' \oplus k_{10}' \oplus k_{11}' \oplus k_{12}' \oplus k_{13}' \oplus k_{14}' \oplus \\ \qquad k_{15}' \oplus k_{16}' \oplus k_{xor})/256\end{cases} \tag{13}$$

步骤 5　预迭代乘以逻辑映射以消除瞬态效应并增加安全性；

步骤 6　迭代 $(k_1' + k_2' + k_3' + k_{xor})$ 次后，将 logistic map 迭代 4 mn 次，得到大小为 1×4 mn 的伪随机向量 $\boldsymbol{PV}$；

步骤 7　通过式(14)计算大小为 1×4mn 的规则向量 $\boldsymbol{R}_{DNA}(i)$；

$$\boldsymbol{R}_{DNA}(i) = floor(8 \times \boldsymbol{PV}(i)) + 1 \tag{14}$$

其中，$i = 1,2,3,\cdots,4$ mn。

步骤 8　将图像 $\boldsymbol{P}2$ 的每个元素编码为四进制，对应 4 种核酸 A、C、G 和 T，得到大小为 1×4 mn 的 DNA 图像 $\boldsymbol{P}3$。

2.3　置换

置换操作交换了普通图像中像素的位置、干扰，降低相邻像素值的高相关性。DNA 水平像素置换的基本步骤：

步骤 1　输入大小为 1×4 mn 的 DNA 图像 $\boldsymbol{P}3$；

步骤 2　根据 k_{xor} 值和密钥 K' 计算 logistic map 的控制参数 u 和初始值 $x(0)$，式(15)：

$$\begin{cases}u = 3.89 + (k_{17}' \oplus k_{18}' \oplus k_{19}' \oplus k_{20}' \oplus k_{21}' \oplus \\ \qquad k_{22}' \oplus k_{23}' \oplus k_{24}' \oplus k_{xor})/256 \times 0.01\\ x(0) = (k_{25}' \oplus k_{26}' \oplus k_{27}' \oplus k_{28}' \oplus k_{29}' \oplus k_{30}' \oplus \\ \qquad k_{31}' \oplus k_{32}' \oplus k_{xor})/256\end{cases} \tag{15}$$

步骤 3　$(k_{17}' + k_{18}' + k_{19}' + k_{xor})$ 乘以逻辑系统，以消除瞬态效应并增加安全性；

步骤 4　迭代 $(k_{17}' + k_{18}' + k_{19}' + k_{xor})$ 次后，将 logistic 系统迭代 4 mn 次，得到大小为 1×4 mn 的伪随机向量 $\boldsymbol{PV}$；

步骤 5　对 $\boldsymbol{PV}$ 元素进行升序排序，f_{PV} 是 $\boldsymbol{PV}$ 序列的新序列，l_{PV} 是 f_{PV} 的索引值；

步骤 6　通过公式(16)对向量 $\boldsymbol{P}3$ 的位置进行打乱。

$$\boldsymbol{P}_4(i) = P_3(l_{PV}(i)) \tag{16}$$

其中，$i = 1,2,3,\cdots,4$ mn。

2.4　DNA 级扩散

在图像加密中，对像素数据的扩散是提高安全性的最重要步骤。DNA 水平扩散步骤：

步骤 1　输入大小为 1×4 mn 的置换 DNA 图像 $\boldsymbol{P}4$ 和大小为 1×4 mn 的关键 DNA 图像 KDNA；

步骤 2　根据 k_{xor} 值和密钥 K' 计算 logistic map 的控制参数 u 和初始值 $x(0)$，公式(17)：

$$\begin{cases}u = 3.89 + (k_1' \oplus k_3' \oplus k_5' \oplus k_7' \oplus k_9' \oplus k_{11}' \oplus \\ \qquad k_{13}' \oplus k_{15}' \oplus k_{xor})/256 \times 0.01\\ x(0) = (k_2' \oplus k_4' \oplus k_6' \oplus k_8' \oplus k_{10}' \oplus k_{12}' \oplus \\ \qquad k_{14}' \oplus k_{16}' \oplus k_{xor})/256\end{cases} \tag{17}$$

步骤 3　预迭代 $(k_1' + k_3' + k_5' + k_{xor})$ 乘以 logistic 系统，以消除瞬态效应并增加安全性；

步骤 4　迭代 $(k_1' + k_3' + k_5' + k_{xor})$ 次后，将 logistic 系统迭代 4 mn 次，得到大小为 1×4 mn 的伪随机向量 $\boldsymbol{PV}$；

步骤 5　通过式(18)计算大小为 1×4mn 的向量 $\boldsymbol{OP}$：

$$\boldsymbol{OP}(i) = floor(7 * \boldsymbol{PV}(i)) + 1 \tag{18}$$

其中，$i = 1,2,3,\cdots,4$ mn。

步骤 6　通过公式(19)对置换后的 DNA 图像 $\boldsymbol{P}4$ 和密钥 DNA 图像 KDNA 进行 DNA 操作。

$$\boldsymbol{P}_5(i)=\begin{cases}f_{+}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==1\\ f_{-}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==2\\ f_{\oplus}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==3\\ f_{\odot}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==4\\ f_{\times}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==5\\ f_{\rightarrow}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==6\\ f_{\leftarrow}(P_4(i),K_{DNA}(i)) & \text{If } OP(i)==7\end{cases} \tag{19}$$

其中 $i = 1,2,3,4,5,6,\cdots,4$ mn。

2.5　扩散 DNA 图像的 DNA 解码

在 DNA 解码步骤中，扩散 DNA 图像 $\boldsymbol{P}6$ 转换为

大小为 $m \times n$ 的灰度密码图像。DNA 解码步骤如下：

步骤 1 输入大小为 1×4 mn 的扩散 DNA 图像 ***P***5；

步骤 2 根据 k_{xor} 值和密钥 K'，计算 logistic map 的控制参数 u 和初始值 $x(0)$。如式(20)：

$$\begin{cases} u = 3.89 + (k_1' \oplus k_2' \oplus k_3' \oplus k_4' \oplus k_5' \oplus k_6' \oplus \\ \quad k_7' \oplus k_8' \oplus k_{xor})/256 \times 0.01 \\ x(0) = (k_9' \oplus k_{10}' \oplus k_{11}' \oplus k_{12}' \oplus k_{13}' \oplus k_{14}' \oplus \\ \quad k_{15}' \oplus k_{16}' \oplus k_{xor})/256 \end{cases} \tag{20}$$

步骤 3 预迭代 $(k_1' + k_2' + k_3' + k_{xor})$ 乘以 logistic map 以消除瞬态效应并增加安全性；

步骤 4 迭代 $(k_1' + k_2' + k_3' + k_{xor})$ 次后，将 logistic map 迭代 4 mn 次，得到大小为 1×4 mn 的伪随机向量 ***PV***；

步骤 5 通过公式(21)计算大小为 1×4 mn 的规则向量 $\boldsymbol{R}_{DNA}$；

$$\boldsymbol{R}_{DNA}(\mathrm{i}) = \mathrm{floor}(8 * \mathrm{PV}(\mathrm{i})) + 1 \tag{21}$$

其中 $i = 1,2,3,4,5,6,\cdots,4\ mn$ 。

步骤 6 根据式(21)，将扩散后的 DNA 图像 ***P***5 解码为解码后的 DNA 图像 ***P***6；

步骤 7 将解码后的 DNA 图像 ***P***6 的每一个核酸 A、C、G 和 T 解码成四进制数字，得到大小为 1×4 mn 的加密四进制图像 ***P***7；

步骤 8 将加密的四元图像 ***P***7 像素值每 4 位编码为 0~255 的整数值，然后转换为大小为 1×mn 的灰度密码图像 ***P***8；

步骤 9 将大小为 1×mn 的灰度密码图像 ***P***8 重塑为大小为 $m \times n$ 的灰度密码图像。

2.6 解密算法

图像解密过程类似于图像加密过程，是使用密钥的图像解密过程的逆向版本。解密算法的总体步骤：

步骤 1 计算密钥 K'，通过外部密钥 K 和哈希值 H 得到 k_{xor} 值；

步骤 2 将密码图像 ***P***8 编码为四进制图像 ***P***7；

步骤 3 将四元图像 ***P***7 编码为 DNA 图像 ***P***6；

步骤 4 计算 key DNA 图像 KDNA；

步骤 5 通过在 DNA 图像 ***P***6 和密钥 DNA 图片 KDNA 之间执行 DNA 扩散的逆运算来消除扩散的影响，并获得非扩散的 DNA 图像 ***P***5；

步骤 6 通过对未扩散的 DNA 图像 ***P***5 进行 DNA 排列的逆运算，消除排列的影响，得到未排列的 DNA 图像 ***P***4；

步骤 7 将非置换 DNA 图像 ***P***4 解码为普通 DNA 图像 ***P***3；

步骤 8 将纯 DNA 图像 ***P***3 解码为纯四元图像 ***P***2；

步骤 9 将普通四元图像 ***P***2 解码为普通图像 ***P***1。

3 实验结果和安全性分析

本文的实验环境软件为 Matlab R2018a 与 Windows 10 操作系统，硬件为 16.0 GB RAM AMD Ryzen 7 5800H with Radeon Graphics 3.20 GHz。选取尺寸为 512×512 的灰度图像“Lena”、“狒狒”作为普通图像。在实验中，将所提出的图像加密方案与其他几种图像加密方案进行了比较。

当外部密钥 key = 6b679b3c77826d30a79e612114a8c18df984c176f4e529f684748ad052241b17 时，本文提出的图像加密方案的仿真结果如图 1 所示，可以看到密码图像与类噪声图像相似，任何关于普通图像的有用信息都不能从他们身上找到。

(a) Lena 的简单图像 (b) lena 对应的图像 (c) Lena 的解密图像

(d) 狒狒的简单图像 (e)狒狒对应的加密图 (f) 狒狒对应的解密图像

图 1 图像加密方案的仿真结果

Fig. 1 Simulation results of the proposed image encryption scheme

3.1 密钥安全分析

加密图像方案最重要的是密钥。密钥空间是可用于加密算法的所有可能密钥的集合，显然密钥空间越大，加密图像算法越安全。当密钥空间大于 $2^{100} \approx 10^{30}$ 时，图像加密算法将能够抵抗暴力攻击。本文的图像加密方案中，可以用密钥解密的密码图像由两部分组成，即 256 位外部密钥和 256 位哈希值。密钥空间大小为 2^{512}，远大于 2^{100}。因此，本文的图像加密方案具有足够大的密钥空间，这将导致抵抗更高的安全级别的暴力攻击。

3.2 直方图分析

图像直方图表示每个灰度强度级别的像素数。

当密码图像直方图接近于均匀分布时，图像加密方案对统计攻击的鲁棒性更强。本文提出的图像加密方案的直方图分析结果如图 2 所示，可见密码图像相邻像素的相关性显著降低，这使得统计攻击更加困难。

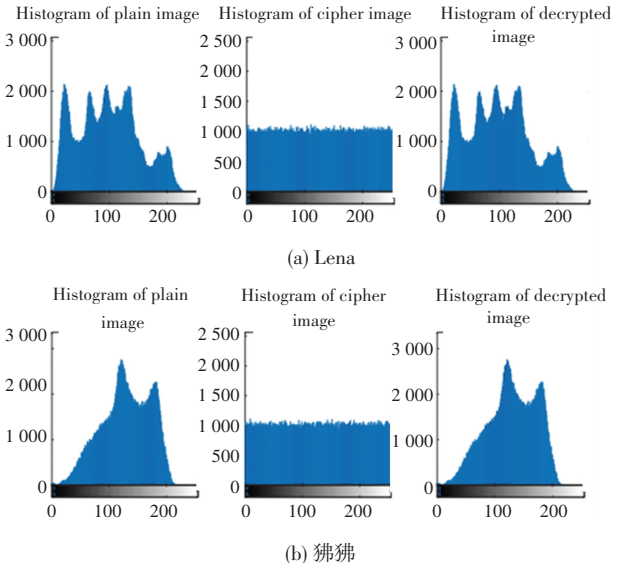

图 2　直方图分析

Fig. 2　Histogram analysis

3.3　相关系数分析

在数字图像中，像素与像素的联系由相关系数来衡量。对于普通图像来说，垂直、水平以及对角方向这 3 个位置的相邻像素之间有着很强的相关性。图像加密就应该降低密码图像相邻像素之间的相关性，以抵抗统计攻击。相关性的理想值为 0。在本文中，从普通图像和密码图像中随机选择垂直、水平和对角线方向上的 10 000 对相邻像素，并通过公式(22)计算两个相邻像素的相关系数：

$$\begin{cases} r_{xy} = \dfrac{\operatorname{cov}(x,y)}{\sqrt{D(x)}\sqrt{D(y)}} \\ \operatorname{cov}(x,y) = \dfrac{1}{N}\sum_{i=1}^{N}(x_i - E(x))(y_i - E(y)) \\ D(x) = \dfrac{1}{N}\sum_{i=1}^{N}(x_i - E(x))^2 \\ E(x) = \dfrac{1}{N}\sum_{i=1}^{N}x_i \end{cases} \tag{22}$$

其中，N 是像素对的数量；x 和 y 是两个相邻像素的灰度值；$E(x)$ 是均值；$D(x)$ 是方差；$cov(x, y)$ 是协方差。

普通图像 Lena 和密码图像 Lena 以及普通狒狒图像和密码狒狒图像的两个相邻像素的相关结果如图 3 所示，Lena 密码图像中两个相邻像素的相关性显著降低。

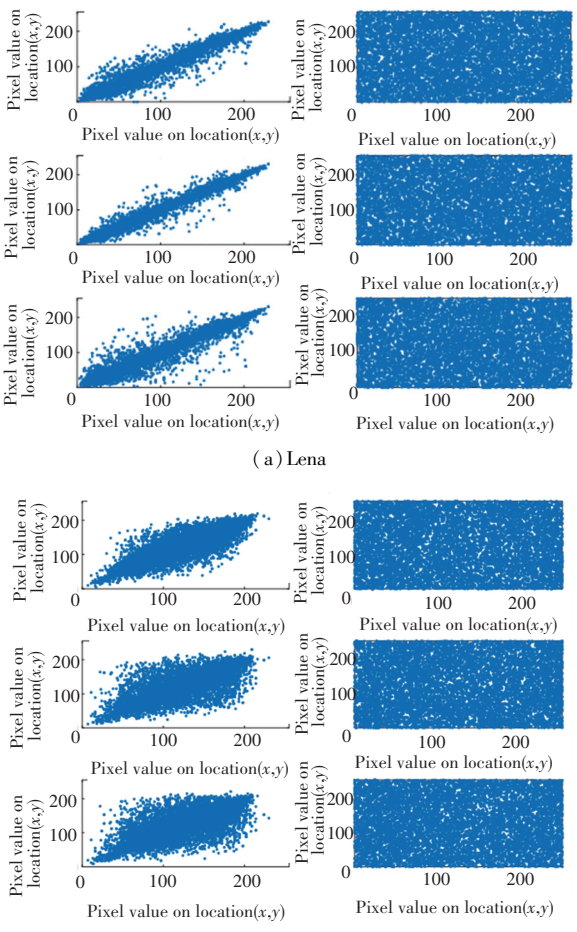

图 3　相邻像素的相关性

Fig. 3　Correlation of neighboring pixels

实验结果表明，“Lena”、“狒狒”这两张图像的相关系数非常高，而相关系数对应的密码图像的系数接近于 0。将本文提出方案与混沌和 DNA 序列循环运算、混沌和 DNA 编码、DNA 互补规则和混沌映射进行对比，图像的相关性分析见表 3，可以发现本文的加密方案有效。

3.4　信息熵分析

信息熵是衡量消息随机性的最重要标准，公式(23)：

$$H(m) = \sum_{i=0}^{2^l-1} p(m_i)\log_2\frac{1}{p(m_i)} \tag{23}$$

其中，l 是像素值的长度，$p(m_i)$ 是信息 m 中符号 m_i 的概率。

对于 256 级灰度的随机灰度图像，信息熵的理论值为 8 。4 个普通图像“Lena”、“Baboon”的信息熵以及相应的密码图像与混沌和 DNA 序列循环运算、混沌和 DNA 编码、DNA 互补规则和混沌映射、DNA 序列操作和混沌系统进行比较见表 4，可见密码图像的所有熵值都非常接近理论值 8，并且密码

图像具有良好的随机分布。因此,信息泄漏的概率可以忽略不计,本文的加密方案对熵攻击具有很强的抵抗力。

表 3 图像的相关性分析

Tab. 3 Correlation analysis of images

测试	位置	普通	本文加密	混沌和 DNA 序列循环运算	混沌和 DNA 编码	DNA 互补规则和混沌映射
Lena	垂直	0.979 4	0.003 7	−0.019 6	−0.004 7	−0.000 9
	水平	0.987 5	−0.000 4	0.019 7	−0.010 6	0.017 1
	对角线	0.967 1	−0.037 8	0.008 8	0.010 5	−0.009 4
狒狒	垂直	0.873 5	−0.011 8	0.001 4	−0.007 5	−0.009 1
	水平	0.787 6	0.012 4	0.011 9	−0.013 6	0.010 1
	对角线	0.757 6	−0.021 5	−0.005 5	0.015 6	−0.004 6

表 4 图像的信息熵分析

Tab. 4 Information entropy analysis of images

测试	输入	本文加密	混沌和 DNA 序列循环运算	混沌和 DNA 编码	DNA 互补规则和混沌映射	DNA 序列运算和混沌系统
Lena	7.592 9	7.999 3	7.999 2	7.999 3	7.995 1	7.998 2
狒狒	7.357 9	7.999 3	7.999 4	7.999 4	7.990 4	7.999 2

3.5 差分攻击

一个安全的图像加密方案应该对纯图像非常敏感。如果明文图像中一个像素的微小变化可以导致密码图像的显著变化,那么加密方案将抵抗差分攻击。像素数变化率 (*NPCR*) 和统一平均变化强度 (*UACI*) 是差异攻击分析的重要标准,由公式(24)定义:

$$\begin{cases} NPCR(C_1,C_2)=\dfrac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}D(i,j)\times 100\% \\ UACI(C_1,C_2)=\dfrac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\dfrac{|C_1(i,j)-C_2(i,j)|}{255}\times 100\% \\ D(i,j)=\begin{cases}0 & C_1(i,j)=C_2(i,j)\\ 1 & C_1(i,j)\neq C_2(i,j)\end{cases} \end{cases} \tag{24}$$

其中, M 和 N 分别是普通图像的宽度和高度, $C_1(i,j)$ 和 $C_2(i,j)$ 是在原始图像的位置 (i,j) 处改变像素值前后的密码图像。

NPCR 和 UACI 的预期值分别为 99.609 4% 和 33.463 5%。随机改变图像“Lena”、“狒狒”10 次,并计算 *NPCR* 和 *UACI* 的最小值、最大值和平均值,结果见表 5、表 6,本文的图像加密方案 *NPCR* 和 *UACI* 结果非常接近预期值。因此,本文提出的图像加密方案对普通图像非常敏感,可以有效抵抗差分攻击。

3.6 密钥敏感度分析

密钥敏感度分析指的是加解密过程中,初始密钥发生微小的变化,经密钥序列发生器或迭代函数作用后所产生的密钥发生巨大变化而进行的分析。因此,在分析过程中用极小差别的密钥来观测最终所呈现的结果。

一个健壮的加密算法应该对其密钥的变化极为敏感,这意味着只有使用正确的密钥,才能正确解密原始图像。为了分析关键灵敏度,进行了如下的测试。首先,随机生成一个 256 位的密钥 *K*1;其次,随机更改其 256 位中的一个,得到两个密钥 *K*2 和 *K*3。分别使用密钥 *K*1 和 *K*2 对 Lena 的普通图像 ***P*** 进行加密,并生成两个密码图像 *C*1 和 *C*2。两个密码图像 *C*1 和 *C*2 完全不同。

表 5 普通图像单像素变化的 *NPCR* 和 *UACI* 值

Tab. 5 *NPCR* and *UACI* values of single pixel variation of ordinary image

测试		本文	混沌和 DNA 序列循环运算	混沌和 DNA 编码	DNA 互补规则和混沌映射	DNA 序列运算和混沌系统
Lena	NPCR	0.996 1	0.996 3	0.996 1	0.995 9	0.996 2
	UACI	0.335 0	0.334 8	0.334 7	0.323 9	0.334 8
狒狒	NPCR	0.996 1	0.996 1	0.996 1	0.996 0	0.996 2
	UACI	0.334 7	0.334 6	0.333 9	0.324 3	0.335 1

表 6　普通图像中两个像素点变化的 *NPCR* 和 *UACI* 值

Tab. 6　*NPCR* and *UACI* values of two pixel changes in a normal image

测试		本文	混沌和 DNA 序列循环运算	混沌和 DNA 编码	DNA 互补规则和混沌映射	DNA 序列运算和混沌系统
Lena	NPCR	0.996 1	1.144 4e−05	0.995 9	0.996 3	0.996 2
	UACI	0.334 6	5.385 5e−07	0.334 7	0.325 9	0.334 4
狒狒	NPCR	0.996 1	1.525 9e−05	0.996 2	0.995 8	0.996 2
	UACI	0.335 1	2.094 3e−06	0.335 2	0.324 0	0.334 9

使用不正确的密钥 *K*2 和 *K*3 的解密图像 *C*1 是有噪声的图像，并且也是完全不同的。实验结果表明，密钥的微小变化将导致完全不同的加密结果。因此，本文的加密方案对密钥非常敏感。

4　结束语

本文提出了一种基于 DNA 计算、混沌系统和散列函数的混合模型的新型图像加密方案，使用来自普通图像和密钥的混合 SHA256/MD5 哈希，通过仅翻转普通密钥的一位来确保混沌系统的初始条件和控制参数发生变化；基于 DNA 序列定义了两个新的代数法则，即 DNA 左位移和 DNA 右位移。同时对其安全性、直方图、相关系数、信息熵、差分攻击进行了分析，证明方案的合理性。实验结果表明，本文提出的图像加密方案有比其他几种具有代表性的图像加密方案更好的安全效果。

参考文献

[1] 陈虹，赵菊芳，郭鹏飞，等. 基于混沌映射的分块循环 DNA 图像加密算法[J]. 计算机应用研究，2022，39(6)：1865−1871.

[2] 周红亮，刘洪娟. 结合 DNA 编码的快速混沌图像加密算法[J]. 东北大学学报(自然科学版)，2021，42(10)：1391−1399.

[3] 司宇晨，滕琳，孟娟. 随机弹射结合 DNA 编码的图像加密算法[J]. 软件导刊，2021，20(8)：185−190.

[4] 陈森，薛伟. 基于混沌和 DNA 随机编码的彩色图像加密算法[J]. 传感器与微系统，2021，40(8)：144−147，156.

[5] 袁立，谢俐，龙颖，等. 基于哈希和 DNA 编码的彩色图像混沌加密算法[J]. 重庆大学学报，2021，44(7)：55−63.

[6] 赵孔文. 基于混合混沌系统和动态 DNA 编码的图像加密[D]. 南昌：南昌大学，2021.

[7] 周辉. 基于混沌的彩色图像加密算法研究[D]. 太原：太原理工大学，2021.

[8] 陈忠仁，张欣，陈健. 基于混沌系统和 DNA 编码的彩色图像加密算法研究[J]. 软件，2020，41(12)：81−88.

(上接第 89 页)

[8] HE K，ZHANG X，REN S，et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016：770−778.

[9] MA Jun，GE Cheng，WANG Yixin，et al. COVID−19 CT Lung and Infection Segmentation Dataset (Verson 1.0) [DS]. 2020. Zenodo.https://doi.org/10.5281/zenodo.3757476.

[10] ZHU J Y，PARK T，ISOLA P，et al. Unpaired image−to−image translation using cycle − consistent adversarial networks [C]// Proceedings of the IEEE international conference on computer vision. 2017：2223−2232.

[11] ISOLA P，ZHU J Y，ZHOU T，et al. Image−to−image translation with conditional adversarial networks [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017：1125−1134.

[12] KARRAS T，LAINE S，AILA T. A style − based generator architecture for generative adversarial networks[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019：4401−4410.

[13] KINGMA D P，BA J. Adam：A method for stochastic optimization [J]. arXiv preprint arXiv：1412.6980，2014.

[14] CHEN L C，PAPANDREOU G，KOKKINOS I，et al. Deeplab：Semantic image segmentation with deep convolutional nets，atrous convolution，and fully connected crfs[J]. IEEE transactions on pattern analysis and machine intelligence，2017，40(4)：834−848.

[15] HEUSEL M，RAMSAUER H，UNTERTHINER T，et al. Gans trained by a two time−scale update rule converge to a local nash equilibrium[J]. arXiv Preprint axXiv：1706.08500，2017.