

基于混沌置乱和混沌加密的 DCT 域数字水印算法

邹长华, 谭世恒, 林土胜

(华南理工大学 电信学院, 广东 广州 510641)

摘 要: 把混沌理论应用到图像置乱中, 提出了基于混沌的数字图像置乱算法. 利用混沌加密算法来对水印数据进行加密, 能有效防止攻击者根据统计学方法来检测到水印. 然后, 对载体图像进行 DCT 变换, 并在 DCT 域上嵌入水印. 通过抗攻击仿真实验表明, 该数字水印算法具有较强的抗剪切攻击和噪声攻击能力.

关键词: 图像置乱; 混沌; 数字水印; DCT 变换

中图分类号: TP391

文献标识码: A

文章编号: 1000-7180(2011)05-0058-05

Digital Watermarking Algorithm in DCT Based on Chaotic Scrambling and Chaotic Encryption

ZOU Zhang-hua, TAN Shi-heng, LIN Tu-sheng

(School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510641, China)

Abstract: In this paper, chaos theory is applied to image scrambling, and a new digital image scrambling algorithm based on chaos is proposed. Then chaotic encryption algorithm is used to encrypt watermark, which can prevent the attackers from detecting the watermark according to statistical methods effectively. The carrier image is putted to DCT transform, and the DCT domain is embedded the watermark. The anti-attack experiments show that the proposed algorithm has strong ability in resisting shear attacks and noise attacks.

Key words: image scrambling; chaos; digital watermarking; DCT

1 引言

随着网络技术的高速发展, 大量重要的信息需要网络进行传输, 使得信息安全显得日趋重要. 数字水印和隐写术是信息加密、实现信息安全的重要技术. 数字图像置乱是数字水印技术的重要组成部分. 数字图像置乱是通过对数字图像的位置或灰度等级做变换, 将一幅图像变另一幅没有明显意义的混乱图像. 通过特定的算法可以从混乱的图像中重构出原图像. 非法使用者即使截获了图像, 也看不到图像的本来面目, 并且难以恢复原始图像. 数字图像置乱是数字水印的重要组成部分, 其方法的优劣将直接影响到信息隐藏的性能.

数字图像置乱技术可分为基于图像位置空间、基于图像色彩空间和基于图像频域空间的置乱. 目

前已有很多文献提出了数字图像置乱的算法, 如基于生命游戏^[1], 基于 Arnold 变换^[2], 基于骑士巡游变换^[3], 基本广义 Gray 码变换^[4], 基于 DES^[5], 基于面包师变换^[6], 基于约瑟夫遍历^[7], 基于队列变换^[8], 基于 Zigzag 变换^[9]等等. 为了提高置乱算法的安全性, 不易被破解, 文中利用混沌系统的基本特性, 提出了基于混沌的数字图像置乱算法, 对水印图像进行置乱. 在此基础上, 为了防止攻击者根据统计学方法来检测到水印, 文中还利用混沌加密算法来对水印数据进行加密, 使得水印图像的呈现均匀分布, 以确保水印传输的安全性.

数字水印算法大致可分为二类: 基于空间域和变换域的方法. 这二类方法分别通过改变某些像素的灰度值和变换域某些系数的值来隐藏水印. 与基于空间域的方法相比, 基于空间域的方法能更有效

收稿日期: 2010-08-02; **修回日期:** 2010-10-10

基金项目: 国家自然科学基金项目(60673191); 广州市科技计划项目(2009J1-C401)

地保证水印的不可见性,这是因为在变换域中嵌入的水印信号能量可以分布到空间域的所有像素上,减轻视觉上的差距.基于上述考虑,文中采用在DCT域嵌入水印的方法,实现信息隐藏.

2 数字水印算法

文中提出的数字水印算法总体流程如图1所示.首先,通过混沌置乱算法对数字水印进行置乱;接着利用混沌加密算法来对置乱后的水印进行加密,得到一维的水印序列;然后对载体图像进行DCT变换,并在DCT域上嵌入水印;最后,对其进行DCT逆变换,即可得到含水印的载密图像.下文将对各部分进行详细的描述.

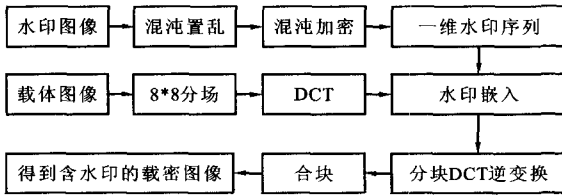


图1 数字水印算法总体流程

2.1 混沌置乱算法

2.1.1 置乱算法的基本方法

混沌是在非线性系统中普遍存在的现象,混沌运动具有对初值的高度敏感性、运动轨迹的遍历性和随机性等特点.产生混沌序列常用的方法有很多,文中采用Logistic方程产生混沌序列. Logistic映射^[10]形式如下:

$$X_{i+1} = \mu X_i (1 - X_i) \quad (i = 0, 1, \dots, k-1) \quad (1)$$

式中, $0 < X_0 < 1$ (且 $X_0 \neq 0.25, 0.5, 0.75$). 取 $\mu = 4$, 则完全处于混沌状态, 且 X_n 在 $(0, 1)$ 内遍历. 由于混沌状态具有对初始值极其敏感的特点, 取不同的初值, 可得到不同轨迹的混沌变量. 对于一幅 m 行 n 列的数字图像, 通过下列规则可以实现空间置乱:

Step1: 初始化产生混沌序列的两个初值 XR_0 和 XC_0 , 行循环和列循环的最大步长 $nRow$ 和 $nCol$, 以及置乱次数 nS .

Step2: 把 XR_0 代入式(1)中, 生成长度为 $m \times nS$ 的混沌序列 XR ; 再把 XC_0 代入式(1)中, 生成长度为 $n \times nS$ 的混沌序列 XC .

Step3: $nRow$ 与混沌序列 XR 相乘, 而 $nCol$ 与混沌序列 XC 相乘, 并对结果进行取整(按四舍五入规则), 分别得到两个新序列 $newXR$ 和 $newXC$.

Step4: 图像进行第 k 次置乱 ($k = 1, 2, \dots, nS$):

图像逐行进行循环移位, 第 i 行从左往右循环移 $newXR[(k-1) \times m + i]$ 位; 接着, 图像逐列进行循环移位, 第 j 列从上往下循环移 $newXC[(k-1) \times n + j]$ 位.

上述的混沌置乱方法, 其密钥有 5 个参数, 即 $Key = (XR_0, XC_0, nRow, nCol, nS)$. 其解置乱过程与置乱过程相似, 在得到密钥 Key 后, 先按上述步骤 Step(1)~(3) 进行, 图像进行第 k 次解置乱 ($k = 1, 2, \dots, nS$): 图像先逐列进行循环移位, 第 j 列从下往上循环移 $newXC[(nS-k) \times n + j]$ 位; 接着, 图像逐行进行循环移位, 第 i 行从右往左循环移 $newXR[(nS-k) \times m + i]$ 位.

下面对置乱算法的安全性进行测试, 测试结果如图2所示. 设定密钥 $Key = (0.2, 0.2, 20, 20, 10)$, 对图2(a) (Lena图, 64×64 的灰度图, 为水印图像) 进行混沌置乱, 置乱结果为图2(b). 若密钥正确, 对图2(b) 进行解置乱得到图2(c); 若 $XR_0 = 0.2 + e^{-12}$, 其他参数正确, 对图2(b) 进行解置乱得到图2(d); 若 $nRow = 21$, 其他参数正确, 对图2(b) 进行解置乱得到图2(e); 若 $nS = 11$, 其他参数正确, 对图2(b) 进行解置乱得到图2(f). 可见只要密钥的其中一个参数出现轻微误差, 得到的图像将与原图相差很大, 这表明置乱算法具有良好的安全性, 不易被解密.

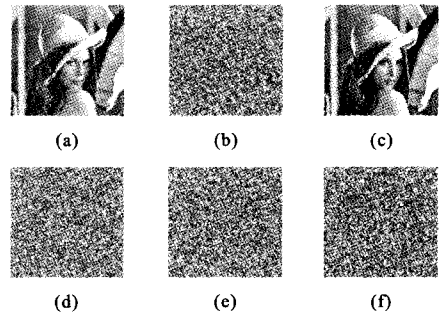


图2 置乱算法的安全性测试

2.1.2 置乱度仿真测试

文中采用文献[7]的置乱度评价方法: 相邻灰度差置乱度. 图像中某像素与其相邻像素的灰度差为

$$GD(x, y) = \frac{\sum [G(x, y) - G(x', y')]^2}{4} \quad (2)$$

式中, $G(x, y)$ 为坐标 (x, y) 处的灰度值, (x', y') 取值为 $(x-1, y)$ 、 $(x+1, y)$ 、 $(x, y-1)$ 和 $(x, y+1)$. 除去图像边缘上的像素外, 计算图像中其余各个像素与其相邻像素的灰度差, 然后相加平均得到整个图像的平均相邻灰度差为

$$E(GD) = \frac{\sum_{x=2}^{M-1} \sum_{y=2}^{N-1} GD(x,y)}{(M-2)(N-2)} \quad (3)$$

灰度值置乱度定义为

$$GDD = \frac{E'(GD) - E(GD)}{E'(GD) + E(GD)} \quad (4)$$

式中, E 和 E' 分别表示置乱前、后的平均相邻灰度差. 这样定义的 GDD 的取值范围为 $(-1, 1)$, 若置乱度小于 0 则表示置乱效果比原图还差, 当然这种情况较少出现; 若置乱度大于 0, 则表示灰度置乱效果比原图要好, 而且越趋近于 1 越好.

下面对置乱算法的置乱效果进行测试, 测试结果如图 3 所示. 设定基本值: $XR_0 = XC_0 = 0.2$, $nRow = nCol = 20$, $nS = 10$. 改变 $nRow$, 其他参数保持不变, 灰度值置乱度变化如图 3(a) 所示; 改变 nS , 其他参数保持不变, 灰度值置乱度变化如图 3(b) 所示; 改变 XR_0 , 其他参数保持不变, 灰度值置乱度变化如图 3(c) 所示. 可见, 增大 $nRow$ 、 $nCol$ 或 nS 能提高灰度值置乱度; 而改变 XR_0 或 XC_0 , 灰度值置乱度会增大或者减小.

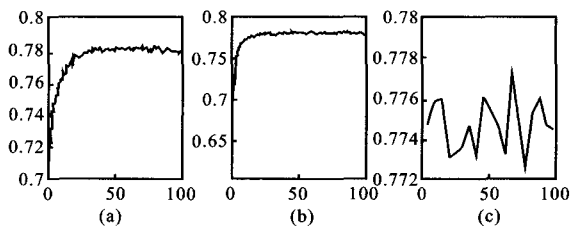


图 3 置乱算法的置乱效果测试

2.2 混沌加密算法

文献[11]提出了一种基于改进型 Logistic 映射的伪随机数发生器, 形式如下:

$$X_{i+1} = 1 - 2(X_i)^2 \quad (i = 0, 1, 2, \dots) \quad (5)$$

$$k_i = \begin{cases} 1, & \text{if } \text{sgn}(X_i) = 1 \\ 0, & \text{if } \text{sgn}(X_i) = -1 \end{cases} \quad (i = 0, 1, 2, \dots) \quad (6)$$

其中式(5)为改进型 Logistic 混沌映射, $\text{sgn}()$ 是取符号函数, $0 < X_0 < 1$ (且 $X_0 \neq 0.25, 0.5, 0.75$), 式(6)为伪随机数发生器. 该随机数发生器具有良好的随机性和自相关性能, 又具有混沌系统的长周期性和对初始条件和控制参数的极端敏感性, 序列的生成仅需要一个模型和初始值, 就可以产生任意长度的序列.

对于一幅 m 行 n 列的灰度水印图像(假设灰度级为 255), 对其进行混沌加密的具体过程如下:

Step1: 灰度图像的每一像素用 8 位的 2 进制数

表示. 按从左到右和从上到下的顺序, 对灰度图像进行扫描, 可以得到 $m \times n \times 8$ 水印序列 L_1 .

Step2: 设定 X_0 , 根据式(5)、(6), 生成长度为 $m \times n \times 8$ 的伪随机序列 L_2 .

Step3: 序列 L_1 和序列 L_2 对应的每一位进行异或, 即可得到长度为 $m \times n \times 8$ 的混沌加密序列 WL ; 若 WL 的每 8 位合成一个像素值, 则可以得到 $m \times n$ 的混沌加密图像.

混沌解密过程与混沌加密过程一样, 只需按上述方法把混沌加密序列 WL 与伪随机序列 L_2 逐位异或即可.

对混沌加密算法的效果进行测试(取 $X_0 = 0.4$), 测试结果如图 4 所示. 其中图 4(a)是图 2(b)的混沌加密图; 图 4(b)是图 2(a)和图 2(b)的直方图; 图 4(c)是图 4(a)的直方图. 可见, 通过混沌置乱并没有改变水印图像的直方图; 而通过混沌加密可以改变水印图像的直方图, 使得其成均匀分布, 从而躲避攻击者根据统计学方法来检测到水印的可能性.

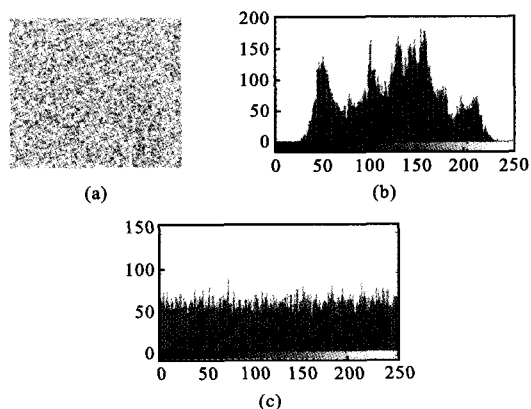


图 4 混沌加密算法的效果测试

2.3 DCT 域数字水印的嵌入与提取

2.3.1 DCT 变换

对 8×8 的像素矩阵 $f(x, y)$, 其二维 DCT(离散余弦变换) 定义为^[12]

$$DCT(i, j) = \frac{1}{4} C(i) C(j) \sum_{x=0}^7 \sum_{y=0}^7 f(x, y) \cos\left(\frac{(2x+1)i\pi}{16}\right) \cos\left(\frac{(2y+1)j\pi}{16}\right) \quad (7)$$

IDCT(离散余弦逆变换) 如下:

$$f(x, y) = \frac{1}{\sqrt{2N}} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} C(i) C(j) DCT(i, j) \cos\left(\frac{(2x+1)i\pi}{2N}\right) \cos\left(\frac{(2y+1)j\pi}{2N}\right) \quad (8)$$

在式(7)、(8)中, $C(x)$ 定义如下:

$$C(x) = \begin{cases} \frac{1}{\sqrt{2}}, & x = 0 \\ 1, & x > 0 \end{cases} \quad (9)$$

这里是先将整体图像分成 8×8 像素块, 然后对 8×8 像素块逐一进行 DCT 变换。

2.3.2 水印的嵌入

水印嵌入的具体算法步骤如下:

Step1: 利用混沌置乱算法, 对水印图像 $W(m \times n)$ 的灰度图) 进行置乱, 水印的置乱图像 W' ; 然后利用混沌加密算法, 对其进行加密, 得到长为 $m \times n \times 8$ 的水印序列 WL。

Step2: 将载体图像 I 按顺序分成多个互不重叠的 8×8 的子块 f_k , 对每个子块进行 DCT 变换得 DCT_k 。子块 f_k 将嵌入水印序列 WL 中的第 k 位。

Step3: 将图像块分成 2 类, 即平滑区和边缘区。在 $DCT(0,0)$ 系数上嵌入水印, 即:

$$DCT'_k(0,0) = \begin{cases} DCT_k(0,0)(1+\alpha) & \text{if } WL(k) = 1 \\ DCT_k(0,0)(1-\alpha) & \text{if } WL(k) = 0 \end{cases} \quad (10)$$

在式(10)中, 在平滑区时, α 不能取得太大, 通常取 0.01, 否则影响载密图像的视觉效果; 而在边缘区时, α 可以取大一些, 通常取 0.02, 以便提高算法的稳健性。

Step4: 对所得结果进行 DCT 逆变换, 再将所有子块合并为整幅图像, 得到含水印的载密图像。

2.3.3 水印的提取

水印提取的具体算法步骤如下:

Step1: 把原始图像和含水印的载密图像分别分割成互不重叠的 8×8 的子块 f_k 和 f'_k 。

Step2: 对上述子块分别进行 DCT 变换得 DCT_k 和 DCT'_k 。

Step3: 按照如下规则提取水印信息, 得到一维的水印序列 WL。

$$WL(k) = \begin{cases} 1 & \text{when } DCT'_k(0,0) \geq DCT_k(0,0) \\ 0 & \text{others} \end{cases} \quad (11)$$

Step4: 根据上文的混沌解密方法, 把一维的水印序列 WL 解密得到水印图像 W' ; 然后根据上文的混沌解置乱方法, 即可得到水印图像 W 。

2.4 数字水印的效果测试

以图 2(a) (Lena 图, 64×64 的灰度图) 为水印图像, 通过混沌置乱和混沌加密可以得到长度为 $64 \times 64 \times 8$ 的一维的水印序列 WL。那么需要载体图像的像素个数至少为 $(64 \times 64 \times 8) \times (8 \times 8)$, 如图 5

(a) (peppers 图, 1024×2048 的灰度图) 所示。图 5(b) 为嵌入水印后的载密图像, 图 5(c) 为从图 5(b) 提取到的水印图。计算载密图像的峰值信噪比 (PSNR) 得到为 42.63, 这表明载密图像与载体图像在视觉上差别很小, 该水印算法具有良好的不可见性。

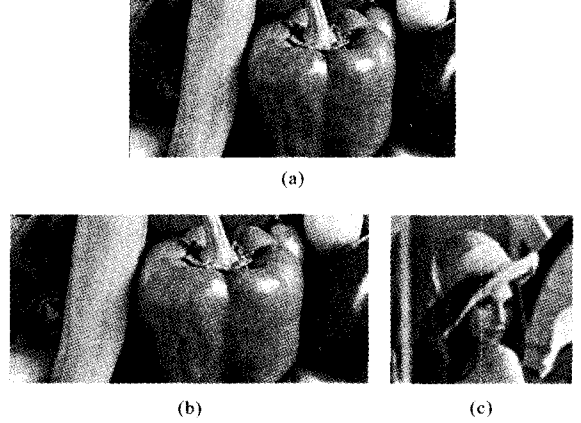


图 5 数字水印算法的效果测试

3 抗攻击实验

数字图像经过信道传输后, 对产生的误码和部分缺损应有较强的鲁棒性, 即能够经得起一定的分析和攻击。该算法的抗剪切攻击效果如图 6 所示。载体图像是图 5(a), 图 6(a) 是图 5(b) 的破损图像, 其中有 $1/4$ 的区域被涂成黑色; 图 6(b) 是对图 6(a) 进行水印提取得到的水印图像。通过对比图 1(a) 和图 6(b) 可以看出, 即使图像有 $1/4$ 遭到破坏, 仍能恢复出较清晰的水印图像, 说明该算法具有较好的抗剪切攻击能力。

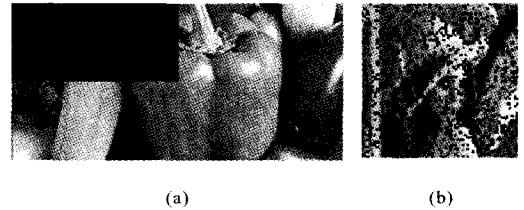


图 6 剪切攻击下的数字水印效果

该算法的抗高斯噪声攻击效果如图 7 所示。图 7(a) 是对图 5(b) 加均值为 0、方差为 0.01 的高斯白噪声后的图像; 图 7(b) 是对图 7(a) 进行水印提取得到的水印图像。通过对比图 1(a) 和图 7(b) 可以看出, 即使图像遭到高斯噪声攻击, 仍能恢复出较清晰的水印图像, 说明该算法具有较好的抗高斯噪声攻击能力。

该算法的抗椒盐噪声攻击效果如图 8 所示。图

8(a)是对图 5(b) 加入椒盐噪声(强度为 5%)后的图像;图 8(b)是对图 8(a)进行水印提取得到的水印图像. 通过对比图 1(a)和图 8(b)可以看出,即使图像遭到椒盐噪声攻击,仍能恢复出较清晰的水印图像,说明该算法具有较好的抗椒盐噪声攻击能力.

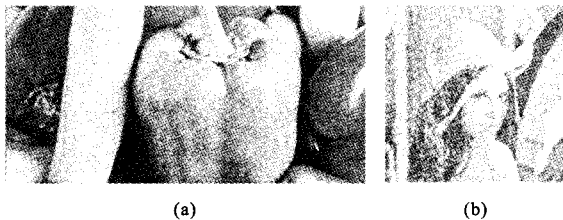


图 7 高斯噪声攻击下的数字水印效果

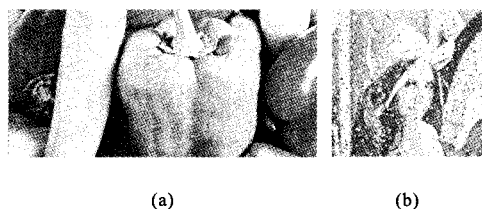


图 8 椒盐噪声攻击下的数字水印效果

4 结束语

文中通过混沌置乱算法对数字水印进行置乱,一方面使得水印图像变一幅没有明显意义的混乱图像,确保其不可见性,另一方面起到加密作用,必须持有正确的密匙 Key 才能获得正确的解置乱图像,即使 Key 有很小的误差,也不能得到正确的解置乱图像. 文中还利用混沌加密算法来对水印数据进行加密,使得水印图像的呈现均匀分布,以躲避攻击者根据统计学方法来检测到水印的可能性. 在前两者的基础上,文中在 DCT 域上实现水印的嵌入. 通过抗攻击实验表明,文中的数字水印算法具有较强的抗剪切攻击和噪声攻击能力.

参考文献:

- [1] 丁玮,阎伟齐,齐东旭. 基于生命游戏的数字图像置乱与数字水印技术[J]. 北方工业大学报,2000,12(1):1-5.
- [2] 丁玮,阎伟齐,齐东旭. 基于 Arnold 变换的数字图像置乱技术[J]. 计算机辅助设计与图形学学报,2001,13(4):338-341.
- [3] 柏森,曹长修,曹龙汉. 基于骑士巡游变换的图像细节隐藏技术[J]. 中国图像图形学报,2001,6(11):1096-1100.
- [4] 邹建成,李国富,齐东旭. 广义 Gray 码及其在数字图像置乱中的应用[J]. 高校应用数学学报: A 辑,2002,17(3):363-370.
- [5] 闫伟齐,邹建成,齐东旭. 一种基于 DES 的数字图像置乱新方法[J]. 北方工业大学学报,2002,14(1):1-7.
- [6] 赵学峰. 基于面包师变换的数字图像置乱[J]. 西北师范大学学报,2003,39(2):26-29.
- [7] 向德生,熊岳山. 基于约瑟夫遍历的数字图像置乱算法[J]. 计算机工程与应用,2005:44-46.
- [8] 李敏,费耀平. 基于队列变换的数字图像置乱算法[J]. 计算机工程,2005,31(1):148-152.
- [9] 冀汶莉,张敏瑞,靳玉萍,等. 基于 Zigzag 变换的数字图像置乱算法的研究[J]. 计算机应用与软件,2009,26(3):71-73.
- [10] Luo J, Shi H. Research of chaos encryption algorithm based on logistic mapping[C]// Intelligent Information Hiding and Multimedia Signal Processing. Pasadena, CA, USA:IEEE, 2006:381-383.
- [11] 刘建夏. 一种混沌伪随机序列的设计及其应用[J]. 计算机工程,2005,31(18):150-152.
- [12] Ali AlHaj. Combined DWT-DCT digital image watermarking[J]. Journal of Computer Science, 2007,3(9):740-746.

作者简介:

邹长华 男,(1986-),硕士研究生. 研究方向为信号处理、图像处理.

谭世恒 男,(1986-),硕士研究生. 研究方向为进化计算、图像处理、模式识别.

林土胜 男,(1945-),教授,博士生导师. 研究方向为电路与系统、信号与信息处理、生物特征识别.