基于 EZW 的图像压缩和树形加密同步算法*

邓海涛† 邓家先 邓小梅

(海南大学信息科学技术学院, 海口 570228)

(2012 年 12 月 6 日收到; 2013 年 1 月 8 日收到修改稿)

为了实现图像的压缩和加密同步提出一种基于内嵌零树小波编码 (embedded zerotree wavelet, EZW) 的压缩和树形加密算法. 加密过程在比特平面编码与熵编码之间进行, 使用密钥对图像压缩产生的上下文和判决进行修正, 实现压缩和加密同步. 对算法进行仿真, 结果表明, 算法具有良好的压缩效率和安全性.

关键词: 图像压缩, 图像加密, 内嵌零树小波编码 (EZW), Logistic 映射

PACS: 07.05.Pj, 05.45.Gg **DOI:** 10.7498/aps.62.110701

1 引 言

对各类电子信息进行加密, 以保证在其处理、存储、传送和交换过程的安全性, 是保证信息安全的有效措施. 同时, 为了减少存储空间、降低传输带宽, 就需要对原始数据进行高效的压缩. Shennon 指出 [1], 冗余度越小, 相关性越小, 不确定度越大, 破译难度越大, 安全性就越高. 传统的先压缩后加密的方法灵活性低, 计算量大, 实时性差. 一种解决方法就是将压缩与加密同步实现. 相对空域加密, 变换域加密具有更强的鲁棒性. 基于离散小波变换 (discrete wavelet transform, DWT) 平台的比特平面编码, 如内嵌零树小波编码 (EZW)[2] 和 JPEG2000 标准 [3] 已被证实能够获得好的图像压缩效果, 且 DWT 变换是对图像的全局变换, 不会出现离散余弦变换 (discrete cosine transform, DCT) 的方块效应. 当前加密领域的研究主要是对空域数据的单纯加密或基于简单的非自适应的熵编码进行加密 [4−8]. 因其所使用的熵编码器模型简单, 算法复杂度较低, 在不进行明文或密文置换条件下, 很容易受到攻击 [6]. 文献 [9] 提出采用分组密码系统在非线性有限域小波变换子带上加密, 但没有研究算法的压缩特性. 文献 [10] 提出一种基于 Logistic 映射的分组加密系统, 将 Logistic 映射产生随机二进制序列分成两部分, 一部分用于对明文的置乱, 另一部分再与置乱明文进行简单的异或运算产生密文, 但在加密过程中没有进行密钥流与明文的相关, 使得该算法在明文攻击时显得很脆弱. 文献 [11] 提出了一种基于多级树集合分割算法 (set partitioning in hierarchical trees, SPIHT) 的感知加密, 通过置换同一父系数的四个子系数的位置进行加密, 但没有考虑子带间相关性, 致使压缩效率不高. 文献 [12] 研究了在小波变换与 SPIHT 编码之间进行加密, 因加密发生在量化和压缩之前, 在有损压缩中鲁棒性极差且影响压缩效率. 本文提出了一种新的简单有效的基于 EZW 的图像压缩和树形加密同步算法. 该算法充分利用 EZW 编码的渐进式传输特性和树形结构, 利用 Logistic 映射生成的密钥对比特平面编码过程中产生的上下文和系数进行修正, 再采用高压缩率的 MQ 算术编码进行熵编码, 并将密文反馈给 Logistic 映射实现密钥流与明文相关, 因加密发生在比特平面编码与熵编码之间, 无需进行数据置换且不破坏小波子带系数之间的相关性, 故算法对压缩性能无影响, 并能够在实现图像数据压缩的同时, 也实现算术加密. 实验结果表明, 算法具有良好的压缩效率和安全性.

2 EZW 编码简述

Shapiro 提出的内嵌零树小波编码算法 (EZW)[2] 利用子带系数之间的相似性, 取得了高

* 海南省自然科学基金 (批准号: 613155) 资助的课题.

† 通讯作者. E-mail: woshidenghaitao@163.com

效压缩效率, 即将小波变换后各级 HL_K, LH_K, HH_K 子带系数构成一棵树, 如图 1 所示. 编码每次都从最低分辨率系数开始扫描, 如果一棵树的根及其子孙的小波系数的绝对值都小于某个给定的阈值 (threshold, T)(这棵树称为零树), 可以用一个预定义的符号代表整棵树, 从而提高压缩比.

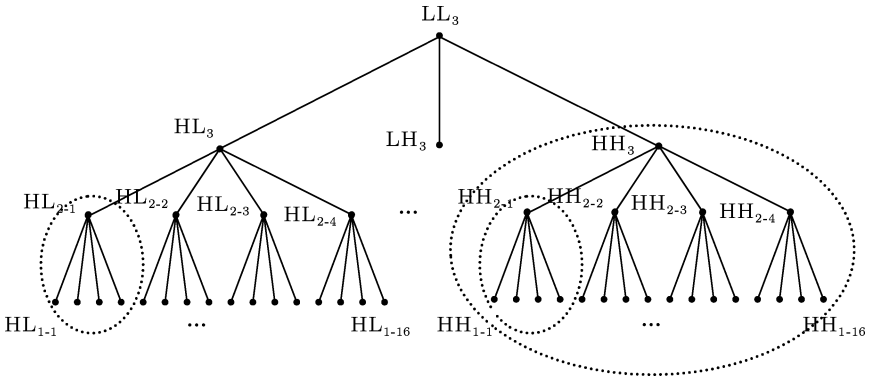

图 1 零树结构

本文算法为每棵树分配一组密钥, 在比特平面扫描过程中, 利用密钥对每个系数产生的上下文 (context, CX) 和判决 (decision, D) 进行修正, 再送往算术编码器进行进一步压缩. 这种编码方式既实现了分辨率压缩, 又达到树形加密的目的.

3 MQ 编码器及加密可行性分析

算术编码的最主要优点是输出的码长能逼近信源的熵, 因此广泛应用于各种压缩算法, 如 JPEG, JPEG2000, JBIG 和 H.264 等. MQ 算术编码是一种基于上下文的自适应二进制算术编码 (context based adaptive binary arithmetic coding, CABAC), 是对无乘法器的 Q 编码算法的改进, 增加了条件交换和概率估计状态机中的贝叶斯学习过程, 采用 Q 编码的位填充策略. 与比特平面编码相结合, 编码从最高有效位 (most significant bit, MSB) 平面开始到最低有效位 (least significant bit, LSB) 平面结束, 按一定的顺序扫描每个位平面, 生成编码数据对 (CX, D), 再经 MQ 编码输出压缩的码流, 如图 2 所示.

图 2 MQ 编码器输入输出模型

MQ 编码器采用一种对原始数据快速自适应概率估计模型. 输入数据流中的信源符号被分成大概率符号 (more probable symbol, MPS) 和小概率符号 (less probable symbol, LPS), 把 LPS 的概率记作 Q_e, CX 用于估计 D 出现的概率. 通过查索引表得到索引 I(CX) 和 MPS, 若 D = MPS, 表明概率估计正确, 进行 CODEMPS 编码, 如 (1) 式所示, 否则表明概率估计错误, 进行 CODELPS 编码, 如 (2) 式所示. 并通过及时的重整化 (renormalization) 操作, 编码间隔 A 始终保持在区间 [0.75, 1.5] 之间, MQ 编码简化模型如图 3 所示.

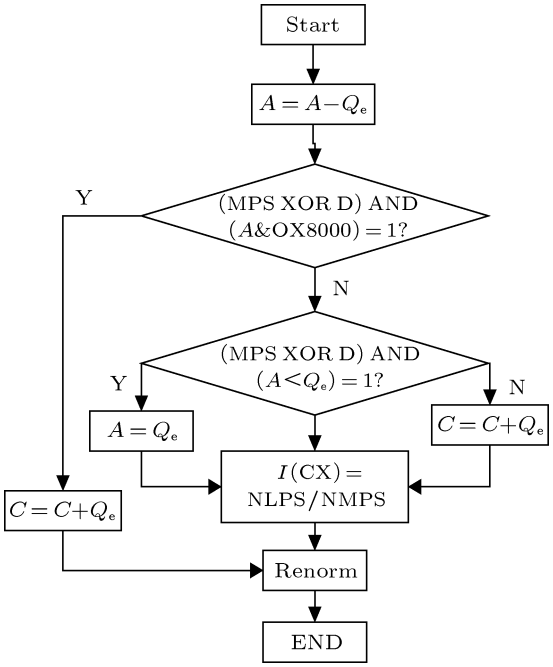

图 3 MQ 编码器算法流程图

因为不同上下文 CX 对应判决 D 的初始分布不完全相同, 而且后续输入判决 D 的条件概率分布也不完全相同. 对于给定的序列, 如果上下文 CX 不

同, 对应的概率子空间也不相同, 编码输出的码字也不相同. 如果改变给定序列中的任何一个上下文 CX 或者判决 D, 就会导致概率子空间的不同, 并会对后续判决的条件概率分布产生影响. 因此, 使用密钥对上下文 CX 或者判决 D 进行修正便可以实现压缩和加密同步进行.

若 $D=\mathrm{MPS}$, 则

$$\begin{aligned} C &= C+(Q_{\mathrm{e}}\times A),\\ A &= A-(Q_{\mathrm{e}}\times A). \end{aligned} \tag{1}$$

若 $D=\mathrm{LPS}$, 则

$$\begin{aligned} C &= C,\\ A &= A\times Q_{\mathrm{e}}. \end{aligned} \tag{2}$$

4 Logistic 映射与密钥的生成

4.1 伪随机序列的产生

混沌现象是非线性动力系统中一种确定性的类随机过程, 该过程具有对初始值敏感、遍历等基本特性, 相对工作在有限离散集上的传统密码系统, 混沌系统工作在无限的连续实数集上, 而这些都是一个好的密码系统所期盼的 [1], 因此被广泛应用于加密技术 [13−16]. 本文使用 Logistic 映射产生混沌轨道 [10,17], 如 (3) 式所示. 其中 μ 为分岔参数, 当 $\mu>3.57$ 时, Logistic 映射处于混沌状态.

$$\mathrm{tau}(x)=\mu x(1-x),\quad x\in[0,1],\quad \mu\in[0,4]. \tag{3}$$

轨道上的每个点采用二进制表示为

$$\begin{aligned} x &= 0.b_1(x)b_2(x)\cdots b_i(x)\cdots,\\ x &\in [0,1]\quad b_i(x)\in\{0,1\}, \end{aligned} \tag{4}$$

其中 $b_i(x)$ 可通过下式得出

$$b_i(x)=\sum_{r=1}^{2^i-1}(-1)^{r-1}\Theta_{(r/2^i)}(x). \tag{5}$$

上式中 $\Theta_t(x)$ 是一个门限函数, 其定义如下

$$\Theta_t(x)=\begin{cases} 0, & x<t,\\ 1, & x\geqslant t. \end{cases} \tag{6}$$

通过式 (3)—(6) 式便可得到具有独立均匀分布的伪随机序列 $B_i^n=\{b_i(\tau^n(x))\}_{n=0}^{\infty}$, 其中 n 是二值序列的长度, $\tau^n(x)$ 是 Logistic 映射第 n 次迭代时的值.

4.2 密钥的生成

在本文算法中, $x_1,x_2,\cdots,x_K,x_{K+1}$ 是以双精度浮点型格式表示的 Logistic 映射的秘密初始值 (K 为小波分解级数), 这些值被用来为图 1 所示的树形结构每一层节点生成一个 32 位二进制密钥 Key. 将 $x_i(i\in[1,K+1])$ 迭代 32 次并得到 32 个数据 $x_{i1},x_{i2},\cdots,x_{i32}$. 利用 4.1 节的方法产生二进制序列 $\mathrm{Key}_i=B_j^1B_j^2\cdots B_j^{32}$.

为了保证加密的安全性及加密和解码双方同步, 本文在加密过程中采用了密文反馈的形式与密钥流相关. 具体实现的方法是对 MQ 编码器每生成一字节的码流, 便用该字节码流对当前密钥重新进行迭代, 以生成新的密钥, 如下式所示:

$$R=32+(C_{\mathrm{num}})\mathrm{mod}32, \tag{7}$$

其中 C_{num} 为当前字节密文, R 为要迭代次数. 迭代完成后得到 R 个数据 $x_{i1},x_{i2},\cdots,x_{iD}$, 从中抽取 32 个数据生成 32 位二进制序列 $\mathrm{Key}_i'=B_j^1B_j^2\cdots B_j^{32}$.

5 图像压缩和树形加密的实现方法

本文提出的基于 EZW 的压缩和树形加密同步算法原理如图 4 所示. 原始图像经 DWT 变换后将原图数据的相关冗余映射成为小波系数的统计冗余, 再进行 EZW 编码. 在比特平面编码过程中生成原始编码数据对 (CX, D), 用密钥 Key 对 CX 和 D 分别进行修正, 产生修正数据对 (CX_1, D_1), 并送往

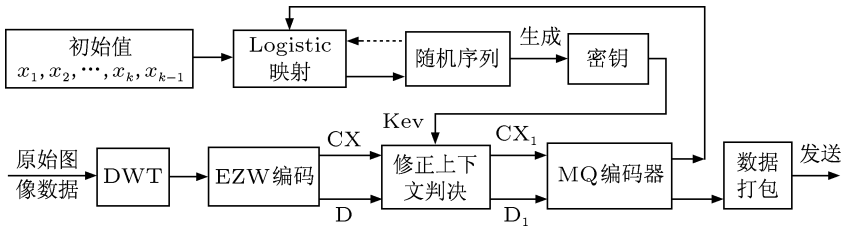

图 4 基于 EZW 图像压缩和树形加密原理框图

MQ 编码器进行熵编码, 最后输出压缩的码流, 从而实现图像压缩和按树形结构加密.

为使 MQ 编码器获得良好的编码效率, 将比特平面编码过程中按所使用的是相邻系数或相邻集合的重要性对产生的判决进行了分类以形成上下文. 由于比特平面编码的判决有两种 (即集合判决和系数判决), 与之相应的将上下文也分为集合上下文和系数上下文两种.

集合上下文的计算, 本文采用简单的同分辨率相邻集合重要性产生, 如图 5 所示, A 表示当前集合重要性, D_0, D_1, D_2, D_3 表示对角相邻集合重要性, H_0, H_1 表示水平方向相邻集合重要性, V_0, V_1 表示垂直方向相邻集合重要性, 它们的取值为 $\{0,1\}$(其中 0 代表集合不重要, 1 代表集合重要). 8 个周边集合共形成 256 种上下文, 经合并形成 4 种集合上下文. 集合上下文 CX_s 的计算公式为

$$\mathrm{CX}_s = (H_0|H_1|V_0|V_1)\times 2 + (D_0|D_1|D_2|D_3), \tag{8}$$

其中 "|" 表示或运算. 显然 CX_s 取值为 $\{0,1,2,3\}$.

D_0	V_0	D_1
H_0	A	H_1
D_2	V_1	D_3

图 5 集合 A 的相邻集合

系数上下文的计算采用最佳截断嵌入码块编码 (embedded block coding with optimized truncation, EBCOT) 中的方法 [18−21] 进一步细化为零编码上下文、幅值上下文、符号编码上下文, 限于篇幅不再赘述.

基于判决修正的算术加密原理如下:

利用密钥对位平面产生的二进制判决进行某种运算, 使得修正后的判决与原始判决不同, 只有当系统编码和解码使用相同的密钥流时才能解码正确, 否则解码错误, 从而实现了判决加密.

设 Key 表示加密密钥流, $D=(d_1,d_2,\cdots,d_N)$ 表示编码产生的原始二进制判决矢量, 长度为 N, 定义一种运算

$$D_1 = f(D,\mathrm{Key}), \tag{9}$$

其中 $D_1=(d_{11},d_{12},\cdots,d_{1N})$ 为修正后的二进制判决矢量, 长度也为 N.

设 Key_1 表示解密密钥流, $\hat{D}=(\hat{d}_1,\hat{d}_2,\cdots,\hat{d}_N)$ 表示解码后的判决矢量, 当 $\mathrm{Key}_1=\mathrm{Key}$ 时, 则 $\hat{D}=D$, 即当解密使用相同密钥流时, 将正确重建原始判决序列, 于是下式成立:

$$\begin{aligned}\hat{D} &= f^{-1}(D_1,\mathrm{Key})\\ &= f^{-1}(f(D,\mathrm{Key}),\mathrm{Key}) = D,\end{aligned} \tag{10}$$

其中 f^{-1} 为 (9) 式对应的逆运算, 所以 (9) 式定义的运算对密钥流应是可逆的.

基于上下文修正的加密方法如下:

设 MQ 编码器的上下文范围为 $(m,m+1,\cdots,m+L_m)$, 设 Key 为加密密钥流, CX 为原始上下文, 取值范围为 $(m,m+1,\cdots,m+L)$, CX_1 为修正后的上下文, 取值范围为 $(m,m+1,\cdots,m+L')$, 上下文修正可以表示为

$$\mathrm{CX}_1 = g(\mathrm{CX},\mathrm{Key},n), \tag{11}$$

其中 $g(\cdot)$ 表示定义的某种变换, n 表示该类上下文出现的顺序, L 与 L' 分别表示 CX 和 CX_1 的种类, 且有 $L<L_m, L'<L_m$. (11) 式需要满足如下要求:

1) 对应给定 CX 和 Key, 对于不同的 n(即在编码的不同时刻), (11) 式运算的修正上下文不能总是固定值, 即 CX_1 是 CX 和 Key 的非线性运算, 否则不能实现算术加密.

2) 在加密和解密过程中, 若比特平面编码和解码生成的上下文 CX 相同, 且使用相同的变换和相同密钥流对 CX 进行修正, 则送往 MQ 编码器的修正上下文 CX_1 相同, 从而得到正确的解码. 即在加密和解密过程使用相同的运算可保证正确解密, 故 (11) 式可以是不可逆运算.

3) 因为 MQ 编码使用的各类上下文范围一定, 如果超出 MQ 编码使用的某类上下文范围, 则进入其他类别的上下文范围. 而不同类别的上下文所对应的初始概率分布不同, 条件概率的跳转规律也不相同, 进行联合压缩加密时, 可能会导致重建图像质量下降. 所以 CX_1 不能超过算术编码所对应类型的范围, 否则可能导致压缩的效率下降.

在本文中, 判决修正采用简单的异或运算, 对密钥流 Key 进行循环移位得到密钥流 Key_1, 即首先利用 Key 的最低位与原始判决进行异或运算, 再将密钥循环移位一次得到 Key_1, 以供下一次判决修正使用.

上下文修正算法是对密钥流 Key 进行循环移位, 取移位后的最低若干位二进制数据 d_k 与 CX(范围为 $(m,m+1,\cdots,m+L)$) 按下式运算:

$$\mathrm{CX}_1 = m + (d_k + \mathrm{CX})\bmod(L_m), \tag{12}$$

其中 mod(·) 表示模运算. CX_1 范围为 $(m, m+1, \cdots, m+L_m)$. 该算法能够满足上文所提的上下文修正的三条要求.

结合 EZW 的比特平面编码, 对整个小波域系数按树形结构扫描顺序进行加密, 能够实现图像压缩和树形加密同步进行.

6 实验结果与分析

实验中采用 (9,7) 不可逆浮点 DWT 变换, 分解级数为 $K=3$, 共需 4 个 Logistic 映射初值 x_1, x_2, x_3, x_4, 分别被用来为图 1 所示的树形结构每一层节点生成一个 32 位二进制密钥 Key, 用其对在比特平面编码过程中生成的数据对 (CX, D) 分别进行修正, 以产生修正数据对 (CX_1, D_1), 并送往 MQ 编码器进行熵编码, 最后输出压缩的码流, 从而实现图像压缩和按树形结构加密同步进行. 随机选择的 4 个初始值如下:

$$x_1 = 0.764350394698857,$$

$$x_2 = 0.689847854354154,$$

$$x_3 = 0.394587025064238,$$

$$x_4 = 0.424789321592047.$$

6.1 重构图像的视觉质量

采用标准 8 位灰度级图像 (Lena, Airplane, Barbara, Boat, Baboon) 进行测试, 在 (12) 式中取 $L = L_m$, 表 1 中显示当图像压缩 8 倍, 输出码率分别为 0.5 bpp, 0.75 bpp, 1.00 bpp, 1.25 bpp, 1.50 bpp 时的测试结果. 图 6 显示了在码率为 0.5 bpp 时重构图像. 结果表明, 在相同码率下, 本文算法与原始 EZW 算法重构图像质量基本相同, 即具有相同的压缩效果.

6.2 密钥空间

一个好的加密算法应该对密钥敏感, 密钥空间应该足够大来抵抗穷举攻击. 本文提出的算法密钥空间大小估计如下:

在 K 级 DWT 分解过程中共得到 $3\times K+1$ 个小波子带. 小波子带的宽度或高度为 $M/2^l$ (M 为原始图像的宽或高, l 为所在的级), 正如在 4.2 节所描述的, 为图 1 中第一棵树的每一层节点分配 32 bit 的密钥作为初始值, 共 $32\times(K+1)$ bit. MQ 编码每输出 1Byte 压缩码流, 便对密钥重新迭代一遍, 即密钥是与密文相关的, 对一幅大小为 $M\times N$ 的图像, 其编码字节数约为 $2^{\lceil \log_2(M\times N)\rceil}$, 所以算法总的密钥空间是 $32\times(K+1)\times 2^{\lceil \log_2(M\times N)\rceil}$ bit. 当 $K=3$, 图像大小为 512×512 时, 密钥空间将达 2^{25} bit. 因此, 算法拥有足够长的密钥空间.

表 1 原始算法与本文算法重构图像质量 PSNR(单位 dB) 比较

码率	原始压缩算法	上下文修正	判决修正	上下文、判决修正
Lena(512×512)				
0.5	36.00	35.98	35.95	35.96
0.75	37.83	37.83	37.71	37.72
1.00	39.17	39.17	39.12	39.12
1.25	40.08	40.07	39.99	40.00
1.50	41.19	41.18	41.07	41.07
Airplane (512×512)				
0.5	35.51	35.47	35.35	35.31
0.75	37.78	37.68	37.47	37.45
1.00	39.97	39.93	39.86	39.84
1.25	41.03	40.96	40.83	40.80
1.50	42.77	42.65	42.44	42.38
Barbara (512×512)				
0.5	30.16	30.15	30.05	30.04
0.75	32.52	32.46	32.28	32.24
1.00	34.96	34.92	34.84	34.81
1.25	36.58	36.50	36.36	36.31
1.50	38.74	38.70	38.65	38.57
Boat (512×512)				
0.5	31.60	31.55	31.40	31.42
0.75	33.95	33.93	33.87	33.86
1.00	35.07	35.05	34.93	34.92
1.25	36.65	36.60	36.43	36.42
1.50	37.65	37.63	37.55	37.55
Baboon (512×512)				
0.5	24.19	24.15	24.06	24.05
0.75	26.33	26.27	26.13	26.10
1.00	27.56	27.52	27.43	27.42
1.25	28.63	28.60	28.48	28.46
1.50	30.04	29.95	29.79	29.75

图 6 视觉质量比较 (a) 原图; (b) 只压缩; (c) 上下文修正; (d) 判决修正; (e) 上下文、判决修正; (f) 解密错误

6.3 抗线性攻击与差分攻击分析

6.3.1 密钥敏感性测试

对密钥作一微小改变以测试密钥敏感性, 即将 Logistic 映射初始值小数点最后 1 位加 1 或减 1, 再进行压缩和加密. 测试中仅将 $x_1 = 0.764350394698857$ 改为 $x_1 = 0.764350394698858$, 其他初始值不变. 然后对密文进行逐位比较, 并计算其 bit 变化百分比. 如表 2 和表 3 所示. 结果表明, 在不同码率下, 位变化百分比都在 50%左右, 这表明密文对密钥是敏感的.

6.3.2 明文敏感性测试

由 (7) 式可知, 参数 R 通过密文反馈与明文相关, 不同的明文使 Logistic 映射在加密过程中迭代不同次数, 从而产生不同的密钥流. 为了测试明文敏感性, 随机选取两幅不同的图像 (限于篇幅本次测试只给出 Lena 图像和 Barbara 图像上下文和判决同时修正的密钥流对比), 进行同步压缩加密, 产生相应的密钥流, 如表 4 所示. 可以看出, 从第二组开始密钥就不相同, 迭代次数 R 也不相同, 所以算法对明文敏感的.

表 2 Lena 图像密钥敏感性测试 (bit 变化百分比)

码率	上下文修正	判决修正	上下文、判决修正
0.50	49.95	49.99	49.98
0.75	50.02	49.98	50.00
1.00	50.01	50.04	49.98
1.25	49.98	50.01	49.95
1.50	50.02	49.99	49.95

表 3 Barbara 图像密钥敏感性测试 (bit 变化百分比)

码率	上下文修正	判决修正	上下文、判决修正
0.50	49.99	49.96	50.03
0.75	49.92	49.98	50.08
1.00	49.96	49.98	49.97
1.25	49.98	49.96	49.92
1.50	50.02	49.99	49.96

以上测试表明, 密文对密钥和明文都很敏感, 从而能够很好的抵抗差分攻击和线性攻击.

6.4 加密与解密速度

表 5 中列出了 Lena 图像在不同码率下的编码和解码时间 (表中 “加密时间” 表示加密时间占总时间的百分比). 从表中可以看出, 随着码率的增大, 总的编解码时间也随之增加. 加密时间占整个算法时间的百分比均小于 50%, 即加密时间不会超过压缩时间, 这种通过牺牲部分运算来达到良好的压缩加密效果是非常值得的.

6.5 与其他压缩和加密算法比较

表 6 中列出了本文算法与文献 [12] 中 SPIHT+SHA-1 同步加密算法及 JPEG+AES 先压缩后加密算法的实验比较结果. 实验结果表明, 当码率较低时, 本文算法重构图像质量要好于 SPIHT+SHA-1 算法和 JPEG+AES 算法, 这主要是因为 SPIHT+SHA-1 算法先对小波系数加密再量化压缩致使各分辨率子带系数相关性被破坏导致压缩效率降低引起; 而 JPEG+AES 算法中主要是 DCT 变换的块效应引起的. 本文算法加密时间占算法总时间百分比较高, 因为本文加密算法是对整个压缩码流进行了加密, 并且密钥流与明文相关.

表 4 Lena 图像和 Barbara 图像部分密钥流

序号	迭代次数	上下文、判决修正	R
Lena 图像对应密钥流			
1	128	9d2e341ae777e5f4ce015f2a61bfbbc5	0
2	176	4228bb63cf28063ce039c45da9bc4dcc	12
3	160	953018ab8251a34397fec9eb5554096a	8
4	128	4d3ec2a5b42ec3eabec065d16a24d665	0
5	224	d18f6bd3cd901d7ef9d8cc5b3b9523ea	24
Barbara 图像对应密钥流			
1	128	9d2e341ae777e5f4ce015f2a61bfbbc5	0
2	224	4245fba5cf44b9a0e0cb2ae5a9e46e95	24
3	224	c11a4db047c4b4b0fb5ebe1952666a35	24
4	208	5d8c119dac0cecd01fd6a31453ec6cf1	20
5	208	c53763f7b7d7b4e7cda772b698b1a325	20

表 5 Lena 图像加密和解密速度

码率	上下文修正		判决修正		上下文、判决修正	
	总时间/s	加密时间 /%	总时间/s	加密时间 /%	总时间/s	加密时间 /%
0.25	1.09	41.41	1.05	38.78	1.11	42.20
0.50	1.14	41.10	1.09	38.52	1.17	42.66
0.75	1.19	43.43	1.14	41.05	1.20	44.14
1.00	1.20	42.89	1.17	41.38	1.23	44.33
1.25	1.27	44.47	1.20	41.56	1.27	44.47
1.50	1.30	45.80	1.25	43.76	1.30	45.76

(表中 “加密时间” 表示加密时间占总时间的百分比)

表 6 本文算法与其他算法重构图像质量 PSNR 的比较

码率	本文算法 (上下文、判决修正)	SPIHT+SHA-1 ($\delta = 1$)	JPEG+AES
重构图像质量 PSNR/dB			
0.25	32.76	29.55	29.06
0.50	35.96	31.13	31.28
1.00	39.12	32.02	38.06
加密时间占总时间的百分比/%			
0.25	42.20	45.87	13.02
0.50	42.66	31.48	8.21
1.00	44.33	21.35	10.24

7 结 论

通过以上分析和仿真表明, 本文提出的压缩和树形加密同步算法密钥流与明文相关, 因加密发生在比特平面编码与熵编码之间, 故算法能够在实现图像数据压缩的同时, 也实现算术加密. 实验结果表明, 加密算法对压缩效率几乎没有影响, 且安全性非常好, 有很大的密钥空间, 对差分攻击和线性攻击具有良好的免疫性具有良好的应用前景.

[1] Shannon C E 1949 *Bell System Technical Journal* **28** 656
[2] Shapiro J M 1993 *IEEE Trans. Signal Processing* **41** 3445
[3] Christopoulos C, Skodras A, Ebrahimi T 2000 *IEEE Trans. Consumer Electronics* **46** 1103
[4] Katti R S, Srinivasan S K, Vosoughi A 2011 *IEEE Trans. Information Forensics and Security* **6** 19
[5] Duan L L, Liao X F, Xiang T 2010 *Acta Phys. Sin.* **59** 6744 (in Chinese) [段黎力, 廖晓峰, 向涛 2010 物理学报 **59** 6744]
[6] Kim H, Wen J T, Villasenor J D 2007 *IEEE Trans. Signal Processing* **55** 2263
[7] Wen J T, Kim H, Villasenor J D 2006 *IEEE Signal Processing Lett.* **13** 69
[8] Bose R, Pathak S 2006 *IEEE Trans. Circuits Syst.* I **53** 848
[9] Chan K S, Fekri F 2004 *IEEE Trans. Signal Processing* **52** 2795
[10] Xiang T, Liao X F 2006 *Physics Letters* A **349** 109
[11] Lian S, Sun J, Wang Z 2004 *IEEE Int. Conf. Multimedia Expo* **3** 2195
[12] Yang H Q, Liao X F, Wong K W, Zhang W, Wei P C 2012 *Acta Phys. Sin.* **61** 040505 (in Chinese) [杨华千, 廖晓峰, Wong K W, 张伟, 魏鹏程 2012 物理学报 **61** 040505]
[13] Zhou W J, Yu M, Yu S M, Jiang G Y, Ge D F 2012 *Acta Phys. Sin.* **61** 080701 (in Chinese) [周武杰, 郁梅, 禹思敏, 蒋刚毅, 葛丁飞 2012 物理学报 **61** 080701]
[14] Liu Q, Fang J Q, Zhao G, Li Y 2012 *Acta Phys. Sin.* **61** 130508 (in Chinese) [刘强, 方锦清, 赵耿, 李永 2012 物理学报 **61** 130508]
[15] He T T, Luo X S, Liao Z X, Wei Z C 2012 *Acta Phys. Sin.* **61** 110506 (in Chinese) [何婷婷, 罗晓曙, 廖志贤, 韦正丛 2012 物理学报 **61** 110506]
[16] Sun K H, He S B, Yin L Z, Duo L K 2012 *Acta Phys. Sin.* **61** 130507 (in Chinese) [孙克辉, 贺少波, 尹林子, 阿地力 · 多力坤 2012 物理学报 **61** 130507]
[17] Yang J Y, Liao X F, Xiao D 2008 *Journal on Communications* **29** 86 (in Chinese) [杨吉云, 廖晓峰, 肖迪 2008 通信学报 **29** 86]
[18] ISO/IEC JTC 1/SC 29/WG1 FCD 14495 public draft, 1997-06. http://www.jpeg.org/public/jpeglinks.htm.
[19] Taubman D 2002 *IEEE Trans. Image Processing* **9** 1151
[20] Taubman D, Ordentlich E, Weinberger M, Seroussi G 2002 *Signal Processing*: *Image Communication* **17** 49
[21] Deng J X, Wu C K, Chen J 2004 *Acta Optica Sinica* **24** 299 (in Chinese) [邓家先, 吴成柯, 陈军 2004 光学学报 **24** 299]

Joint compression and tree structure encryption algorithm based on EZW*

Deng Hai-Tao† Deng Jia-Xian Deng Xiao-Mei

(*School of Information Science and Technology, Hainan University, Haikou* 570228*, China*)

(Received 6 December 2012; revised manuscript received 8 January 2013)

Abstract

A novel joint compression-encryption algorithm based on embedded zerotree wavelet (EZW) coding is proposed. Encryption process is performed before entropy coding. The principles of the context modification and the decision modification and described. Simulation results show that the proposed algorithm has a good effect on security, and has the same compression efficiency compared to the original image compression algorithm.

Keywords: image compression, image encryption, embedded zerotree wavelet (EZW), logistic map

PACS: 07.05.Pj, 05.45.Gg **DOI:** 10.7498/aps.62.110701

* Project supported by the Natural Science Foundation of Hainan Province, China (Grant No. 613155).

† Corresponding author. E-mail: woshidenghaitao@163.com