一种结合 JPEG 压缩编码的彩色图像加密算法*

肖迪† 谢沂均

(信息物理社会可信服务计算教育部重点实验室, 重庆大学计算机学院, 重庆 400044)

(2013 年 7 月 24 日收到; 2013 年 9 月 17 日收到修改稿)

为了研究针对联合图像专家小组压缩标准 (joint photographic experts group, JPEG) 彩色图像特点的加密算法, 综合选择加密和在编码过程中加密两种思路, 提出了一种空域和频域结合的加密算法. 首先在空域对 8×8 分块进行扩散置乱, 然后利用边缘检测手段找出包含细节信息较多的重要分块, 先加密所有分块中的直流 (direct current, DC) 系数, 再选择重要分块中的一部分交流 (alternating current, AC) 系数进行加密, 最后将分块重要性标记信息嵌入 AC 系数中进行传输. 通过理论分析和大量实验证明, 算法格式兼容, 密文图像视觉效果好、色彩分布均匀; 算法密钥空间大, 密钥敏感性强, 安全性良好.

关键词: JPEG 彩色图像, 边缘检测, 选择加密, 结合编码加密

PACS: 05.45.Gg **DOI:** 10.7498/aps.62.240508

1 引 言

随着图像捕捉设备的不断升级以及互联网的持续发展, 高清多媒体图像被越来越广泛地应用, 多媒体图像的安全传输和存储成为一个重要的问题. 加密是一个非常有效的解决方案, 但是对于传统的加密方法如数据加密标准 (data encryption standard, DES), 高级加密标准 (advanced encryption standard, AES), 国际数据加密算法 (international data encryption algorithm, IDEA), RC5 等等, 都是针对一维的数据进行加密, 在实际应用中是不愿意采用的, 因为: 1) 图像数据量很大, 整个图像转换为一维数据单独加密会造成加密效率低下, 并且加密后的冗余数据较多; 2) 加密时间长, 对于要求较高的网络实时传输会造成延迟; 3) 不符合某些特殊环境下的计算要求, 如 Smartphone, 无线传感网络 (WSN) 等.

因此, 为了降低加密数据量以及减少冗余, 针对图像特殊格式的加密逐渐成为研究热点. 而目前主流的研究思路有以下两个: 1) 结合图像编码过程加密 [1−8], 在图像压缩的过程中无缝嵌入加密算法, 使得加密和压缩同时进行, 并且格式兼容; 2) 选择加密 [9−11], 针对于某种特定格式的图像, 选择一部分关键数据进行加密就可加密整幅图像.

联合图像专家小组压缩标准 (JPEG) 是由国际标准化组织 (International Standardization Organization, ISO) 和国际电报电话咨询委员会 (International Telephone and Telegraph Consultative Committee, C-CITT) 建立的第一个国际数字图像压缩标准, 也是目前应用得最广泛的图像压缩格式之一. 利用上述思路针对 JPEG 格式的加密方案有较多的研究成果, 如 Ge 等 [3] 提出一种先以 Logistic 置乱 8×8 大小的像素块, 再加密离散余弦转换 (discrete cosine transformation, DCT) 系数符号的 JPEG 彩色图像算法, 然而单纯的 Logistic 映射的分布不均匀, 其产生序列的安全性不够好. 文献 [5] 首先利用二维耦合映像格子 (2-dimensional coupled map lattices, 2D CML) 生成混沌序列, 然后用混沌序列加密将 JPEG 中直流 (DC) 系数和所有交流 (AC) 系数的符号组成的加密对象集合. 然而在文献 [1] 中给出了对应的破解方法: 将 DC 系数全部置为 128 并且将所有 AC 系数的符号设为正, 有效的明文信息就能被攻击者得到. 文献 [6] 也提到这个问题, 因此这种

* 重庆市杰出青年科学基金 (批准号: CSTC2011JJJQ40001) 资助的课题.

† 通讯作者. E-mail: dixiao@cqu.edu.cn

方式并不安全. Lian 等 [7] 提出了一种结合 JPEG 压缩过程的加密算法, 首先置乱 8×8 图像块, 再将 DC 系数置乱, 最后加密 DCT 系数的符号, 该算法加密效果不够理想, 直接从密文图像中能获取明文色块、模糊的轮廓等明文信息, 安全性不高. Xu 等 [9] 将图像 DCT 并且量化之后每个 8×8 中的 DC 系数和头两个 AC 系数提出来, 用 Chen 系统生成混沌序列进行异或加密, 然而这种方式加密的 DCT 系数太少, 剩余的 AC 系数仍然可能泄露明文信息.

本文利用选择加密的思路, 引入结合压缩过程加密的方式, 提出一种新的结合 JPEG 压缩编码的彩色图像加密算法. 该算法同时具有结合压缩编码加密和选择加密这两种思路的优点. 联合使用两个混沌特性较好的映射作为伪随机数发生器, 较为新颖地利用图像边缘检测的手段找出边缘信息丰富的 8×8 图像块标记为重要块进行重点加密以保证图像细节完全保密, 并且将块的重要性信息嵌入到 AC 系数中. 同时, 本算法相较于上述一些算法具有更好的加密效果, 色彩混淆均匀, 完全无法识别明文图像的任何信息. 最后利用 4 个图像质量评价指标, 通过大量实验证明, 这种加密方式能达到以下效果:

1) 算法是格式兼容的, 加密文件可以正常解码;

2) 色彩混淆均匀, 明文色彩和轮廓信息完全隐藏, 在视觉上完全保密;

3) 利用四个图像质量评价指标测试密文图像质量, 加密效果优于 AES 加密.

2 结合 JPEG 压缩的彩色图像加密算法

本节将描述算法的具体细节和流程. 图 1 是算法的流程图, 灰色方框中的操作表示标准 JPEG 编码过程.

图 1 中, X, Y 是 Henon 映射生成的混沌序列; P^1 和 P^2 是分段线性混沌映射 (piecewise linear chaotic map, PWLCM) 生成的混沌序列; IV_1 和 IV_2 作为 RC5 算法的初始密钥用来加密分块重要性比特序列 S 以及 DCT 系数符号. 密钥的编排以及混沌序列的生成在 2.1 中介绍. 对照图 1, 完整的算法步骤描述如下:

步骤 1 生成混沌序列 (X,Y), P^1 和 P^2 以及 RC5 的初始密钥 IV_1 和 IV_2;

步骤 2 将明文图像利用 Canny 算子进行边缘检测得到边缘图 E;

步骤 3 转换明文图像色彩空间至 YCbCr, 利用边缘图 E 对每个分量的 8×8 分块进行重要性标记;

步骤 4 利用混沌序列 (X,Y), 对分块进行扩散置乱;

步骤 5 执行 DCT 变换、量化过程;

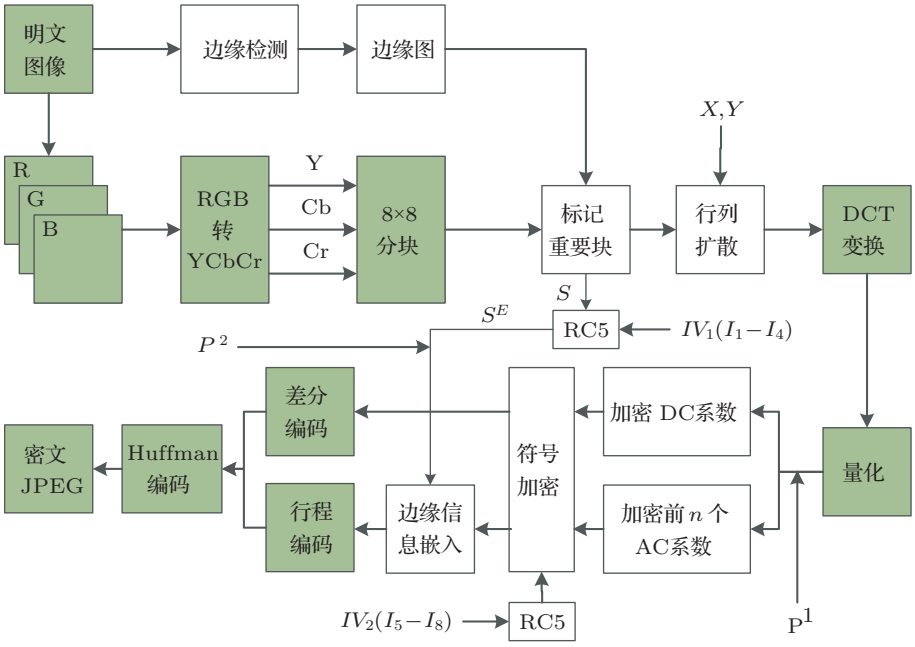

图 1 结合 JPEG 压缩的彩色图像加密算法流程, 灰色方框表示标准 JPEG 编码过程

步骤 6 利用混沌序列 P^1, 对所有 DC 系数和重要块中前 n 个 AC 系数进行异或加密, 并且用 RC5 算法对这些系数的符号进行加密;

步骤 7 将明文边缘信息 (重要块比特序列 S) 用 RC5 加密得到 S^E, 并利用随机位置序列 P^2 将 S^E 嵌入 AC 系数中;

步骤 8 对 Y, Cb, Cr 三个分量执行步骤 3—6 的操作;

步骤 9 完成剩余熵编码步骤, 得到加密的 JPEG 文件.

2.1 密钥的编排及混沌序列的生成

本文引入 Henon 映射和 PWLCM 映射生成混沌序列用于加密算法中. Henon 映射 [12] 是一个离散时间动力系统, 一个典型的 Henon 映射可以用下面的差分方程表示:

$$\begin{cases} x_{n+1} = y_{n+1} + 1 - 1.4x_n^2, \\ y_{n+1} = 0.3x_n, \end{cases} \tag{1}$$

其中 $x \in (-1.5, 1.5)$, $y \in (-0.4, 0.4)$.

PWLCM 映射 [13] 是分段混沌映射, 它相较于一般的混沌映射如 Logistic 映射有更好的动力学和统计学特性, 它可以表示如下:

$$x(k+1) = C[x(k);\mu] = \begin{cases} \dfrac{x(k)}{\mu} & x(k) \in [0,\mu) \\ \dfrac{x(k)-\mu}{0.5-\mu} & x(k) \in [\mu, 0.5) \\ C[1-x(k);\mu] & x(k) \in [0.5, 1) \end{cases}, \tag{2}$$

其中正实数 $\mu \in (0, 0.5)$, $x \in (0, 1)$.

算法共用到 5 个密钥 $(K_1, K_2, K_3, K_4, K_5)$ 和 3 组混沌序列 (X, Y), P^1 和 P^2, 它们的编排与生成如图 2. 其中参数扰动函数定义为 $F(a,b,c) = \gamma \times \dfrac{a+b+c}{3}$, γ 是值域调整因子, 将结果调整到对应参数的范围. 设待加密图像的宽度为 W, 高度为 H, 且 $w = W/8$, $h = H/8$, 对混沌 1, 用 $x_0 = K_1$, $\mu = K_2$ 作为 PWLCM 的初值和参数迭代 1000 次得到初始参数 I_0; 对混沌 2, $\gamma = 0.4$, 用初值 $x_0 = K_3$, $y_0 = F(I_0, K_4, K_5)$ 生成 (X, Y), X, Y 分别迭代 $3 \times w$ 和 $3 \times h$ 次; 对混沌 3, $\gamma = 0.58$, 用初值 $x_0 = K_4$ 和参数 $\mu = F(I_0, K_3, K_5)$ 迭代 $3 \times w \times h \times 17$ 次生成 P^1; 对混沌 4, $\gamma = 0.58$, 用初值 $x_0 = K_5$ 和参数 $\mu = F(I_0, K_3, K_4)$ 迭代 $3 \times w \times h$ 次生成 P^2. IV_1 和 IV_2 分别为 128 位 RC5 密钥. IV_1 由 I_1—I_4 组成, 其中 I_1 是混沌 1 迭代 1001 次的值, I_2 是迭代 1002 次的值, 以此类推. 按照 IEEE 754 浮点数标准将 I_1—I_4 这 4 个 32 位浮点数转换为二进制并连接组成 IV_1, 类似的可以将 I_5—I_8 转换为 IV_2.

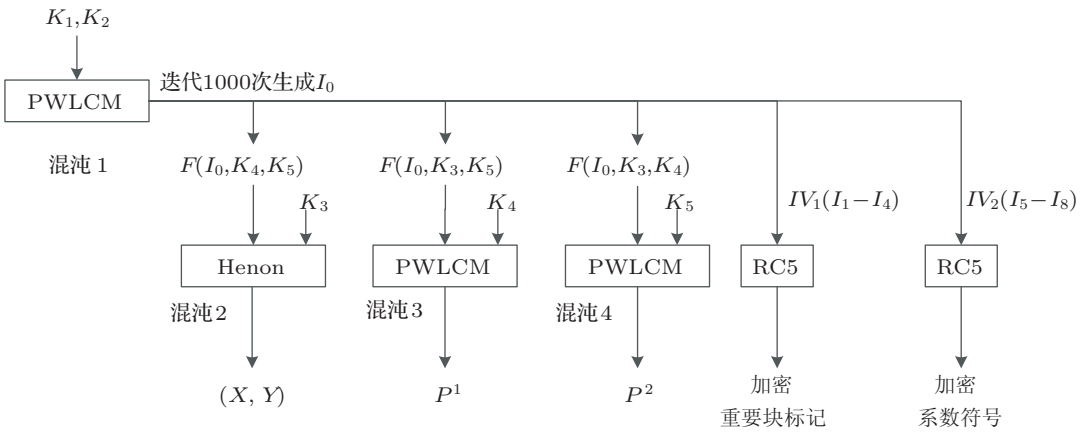

图 2 密钥的编排和混沌序列的生成

2.2 图像边缘检测

对一幅图像加密是将灰度均匀化, 同时打乱像素的位置, 造成人眼无法识别, 这个过程相当于将图像的亮度信息和边缘细节信息进行了隐藏. 对于 JPEG 编码的图像, 目前大多数针对 JPEG 图像频域的加密方式都是对 DC 系数或者 AC 系数进行某种操作来达到加密的目的 [1−3,5−9]. 但是, 加密所有 DCT 系数会造成计算数据量过大. 结合选择加密的思想, 本文利用边缘检测选择重要信息加密: 首先对图像进行边缘检测, 生成一个边缘二值图像, 在进行 8×8 分块后, 边缘像素会对应落在每个分块中. 设一个阈值 T, 如果落在分块中的边缘像素的个数大于等于 T, 此分块记为重要块, 否则为不重

要块. 重要块中含有丰富的细节信息, 如果重要块被加密就能够隐藏整幅图的细节信息.

本文选择 Canny 算子进行图像边缘检测. Canny 算子是满足信噪比准则、定位精度准则和单边缘响应准则最佳的边缘检测算子 [14], 它具有一个参数 T_c, 允许根据特定要求进行调整以识别不同的边缘特性, 一般来说 T_c 越小, 检测出来的边缘越细腻, 包含细节信息越多. 图 3 是利用 Canny 算子对 Lena 进行边缘检测的结果, 参数 $T_c = 0.1$.

2.3 分块行列扩散

分块行列扩散操作可以达到空域置乱的目的. 行列扩散操作是以 8×8 分块为单位, 首先将图像每行分块向右循环移位, 每行分块移动的距离不同; 然后将图像每列分块向下循环移位, 每列移动的距离不同. 如此经过多轮行扩散与列扩散, 图像分块位置被完全加密. 图 4 为分块扩散操作示意图.

利用混沌序列 (X,Y), 用于分块行扩散操作和列扩散操作. 令 W, H 为分别图像的宽度和高度, 则 $w = W/8$, $h = H/8$ 分别表示分块的列数和行数. 首先利用 Henon 映射生成长度为 w 的混沌序列 X 以及长度为 h 的混沌序列 Y; 然后使 $x_i = (x_i \times 1000000) \bmod h \quad (x_i \in X)$, 且 $y_i = (y_i \times 1000000) \bmod w \quad (y_i \in Y)$, 此时 X 和 Y 分别表示行列的移动距离. 对于第 i $(i = 0, 1, \cdots, h-1)$ 行, 将该行向右循环移动 x_i 个分块; 对于第 j $(j = 0, 1, \cdots, w-1)$ 列, 将该列向下循环移动 y_j 个分块. 经过多轮行列扩散操作就可以将分块完全扩散.

图 3 利用 Canny 算子对 Lena 进行边缘检测的结果

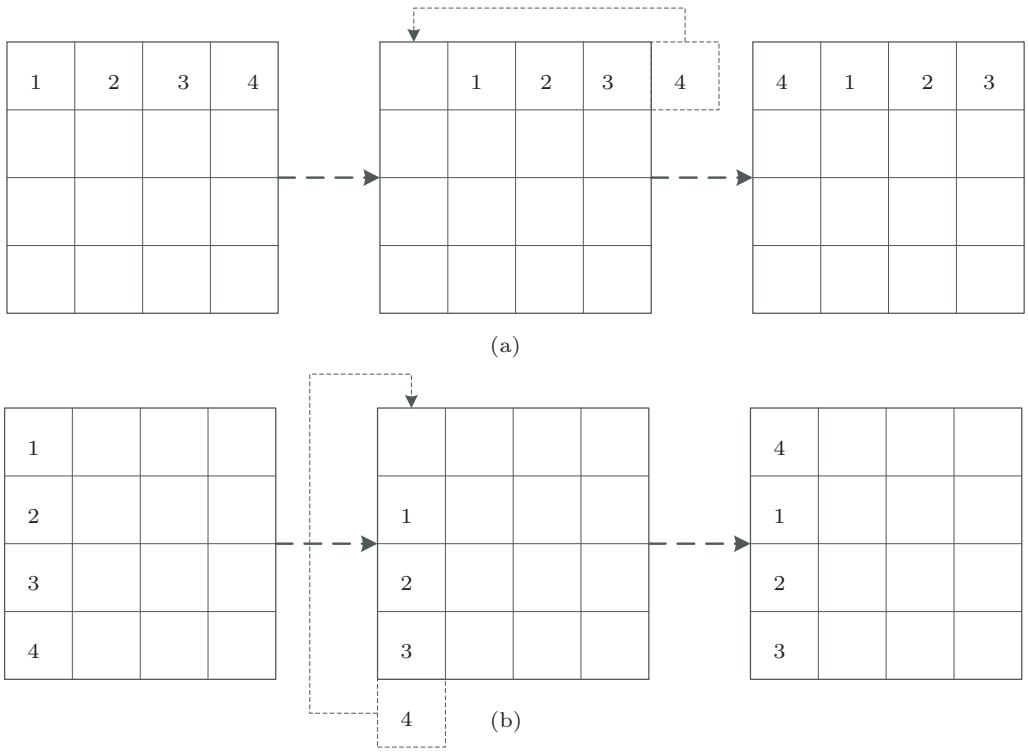

图 4 分块行列扩散操作 (a) 行扩散; (b) 列扩散

2.4 DCT 系数加密

算法将对所有 DC 系数和重要块中的部分 AC 系数加密, 首先采用 PWLCM 映射产生的混沌序列 P^1 与 DC 系数进行异或加密. 对于每个分量, 有 $w \times h$ 个 DC 系数, 遍历所有 DC 系数 d_i, 对 d_i 执行下面操作:

$$d_i = d_i \text{ xor } ((p_i^1 \times 1000000) \bmod (1023/Q_0)), \quad (3)$$

其中 p_i^1 是混沌序列 P^1 中第 i 个值, Q_0 为量化表中最左上角对应 DC 系数的值. DCT 系数值域为 $[-1024, 1023]$, $\bmod(1023/Q_0)$ 保证加密后的 DC 系数没有超出值域范围, 可以进行正常的编解码.

对于重要块的 AC 系数, 加密方式和上面类似. 首先, 对于每个分块按照 Zigzag 顺序选择前 n 个 AC 系数作为被加密对象; 遍历所有 AC 系数 a_i, 对 a_i 进行以下操作:

$$a_i = a_i \text{ xor } ((p_i^1 \times 1000000) \bmod Q_i), \quad (4)$$

其中 p_i^1 为混沌序列 P^1 中第 i 个值, Q_i 为量化表中与 AC 系数位置对应的值. 一般来说 AC 系数取得越多, 加密效果越好, 但是会加重计算负担和编码负担. 经过多次实验, 本文选取 $n = 16$ 作为一个保证良好加密效果和计算量的折中.

上述操作对 Y, Cb, Cr 三个分量分别进行. 并且需要注意的是 (3) 式和 (4) 式中的异或运算都是对 DCT 系数的绝对值进行计算, 待上述计算完成之后, 再将这些系数的符号取出组成符号比特序列, 用 RC5 加密该符号比特序列达到将 DCT 系数信息完全隐藏的目的.

2.5 重要块标记嵌入

每一幅图像都会产生独特的边缘信息, 如果将它放入密钥中进行传输, 那么每加密一幅图像就会传输一个新的密钥用于解密, 而一般加密系统会使用同一个密钥来加密一定数量的内容, 所以上述的做法显然不太合适. 本文将边缘信息嵌入到 AC 系数中来解决这个问题. 在对重要块进行标记之后, 将每一个分块的重要性存到一个比特序列 S 中, 0 表示不重要, 1 表示重要. 将 S 用 RC5 算法加密得到 S^E, S^E 同样为一个比特序列, 此时设计如下方法将 S^E 嵌入 DCT 系数中:

$$a^i = \begin{cases} a^i \times 2 & S_i^E = 0 \\ a^i \times 2 + 1 & S_i^E = 1 \end{cases}. \quad (5)$$

其中 a^i 为第 i 个分块所选择被加密的 n 个 AC 系数的其中一个, 这里 a^i 的选择方法是: 利用混沌序列 P^2, 对于第 i 个分块, 计算 $pos = (p_i^2 \times 1000000) \bmod n$, 也就是选择该分块 n 个 AC 系数中的第 pos 个, 相当于产生一个随机位置, 这样攻击者无法知道嵌入的信息的具体位置, 保证嵌入信息不被窃取. 对于第 i 个分块, 如果对应的 $S_i^E = 0$, 则 a^i 被偶数化; 如果 $S_i^E = 1$, 则 a^i 偶数化之后加 1. 用这种方式嵌入的 S^E 可以在解密的时候完全提取: 获取第 i 个分块第 pos 个 AC 系数 $\hat{a}^i$, 如果 $\hat{a}^i$ 是偶数, 则 $\hat{a}^i = \hat{a}^i/2$, $\hat{S}_i^E = 0$; 如果 $\hat{a}^i$ 是奇数, 则 $\hat{a}^i = (\hat{a}^i - 1)/2$, $\hat{S}_i^E = 1$; 扫描完所有分块得到完整的 $\hat{S}^E$, 用 RC5 解密得到重要性比特序列 $\hat{S}$, 此时 $\hat{S}$ 可以用于后续解密操作. 边缘信息的嵌入节省了密钥存储空间, 加快了传输效率, 并且嵌入信息之后与压缩格式兼容.

算法的解密为上述过程的逆过程: 对每个分量, 首先得到分块重要性, 再恢复 DC 系数和 AC 系数, 重建图像后恢复分块位置就得到解密图像.

3 实验结果

本文选择南加州大学信号与图像处理研究所 (University of Southern California Signal and Image Processing Institute, USC-SIPI) 图像库 [15] 中的彩色图像作为实验对象, 实验环境为 Intel 3.10 GHz CPU, 1 GB RAM, 利用 Matlab2011b 实现算法. 同时为获得压缩效果和高质量图像之间的平衡, 质量因子设为 $Q = 85$. 密文图像在视觉上无法识别, 色彩混淆均匀, 明文色彩和轮廓信息完全被隐藏. 并且通过四个图像质量评价指标测试密文图像质量发现加密效果优于 AES 加密, 并且算法是格式兼容的.

3.1 加密结果及评价

设 $K_1 = 0.36$, $K_2 = 0.38$, $K_3 = 0.50$, $K_4 = 0.30$, $K_5 = 0.40$, 产生混沌序列 (X, Y) 长度分别为 $3 \times h \times 10$ 和 $3 \times w \times 10$, 其中 h, w 分别为分块的行数和列数, 用于 3 个分量执行 10 轮分块扩散置乱; 混沌序列 P^1 的长度为 $3 \times h \times w \times 17$, 用于 3 个分量 17 个系数 (1 个 DC 系数和 16 个 AC 系数) 的异或操作; 混沌序列 P^2 的长度为 $3 \times h \times w$. 图 5 是用本算法对 Lena 和 Baboon 加密之后的结果. 从图中可以看到色彩混淆均匀, 无法识别原图色彩信息或者细节

轮廓, 有效解决了文献 [6] 以及文献 [10] 算法加密结果中能够获取到明文主色调和部分色块的问题, 加密效果更好. 为了进一步验证算法的加密效果, 本文使用以下几个指标来评价密文图像和明文图像之间的差距.

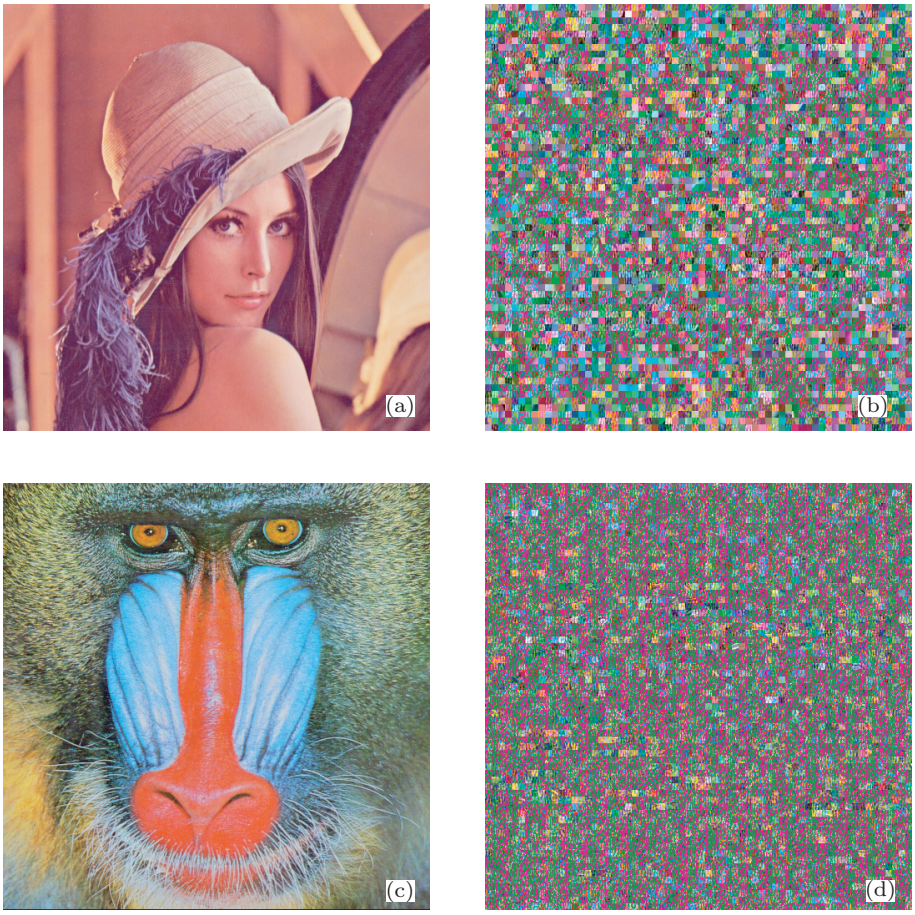

图 5 加密结果 (a) Lena 明文; (b) Lena 密文; (c) Baboon 明文; (d) Baboon 密文

3.1.1 峰值信噪比 (peak signal to noise ratio, PSNR)

PSNR 主要用作测量加密图与原图像之间的差别, 差别越大, PSNR 值越小, 加密效果越好. PSNR 定义如下:

$$\mathrm{PSNR} = 10 \times \lg\left(\frac{255^2}{\mathrm{MSE}}\right), \tag{6}$$

$$\mathrm{MSE} = \frac{1}{wh}\sum_{i=0}^{w}\sum_{j=0}^{h}(I(i,j)-J(i,j))^2, \tag{7}$$

其中图像尺寸为 $w \times h$, $I(i,j)$ 与 $J(i,j)$ 分别代表图像 I 和 J 中 (i,j) 位置的像素值.

3.1.2 平均结构相似指数 (mean structural similarity, MSSIM)

结构相似指数 (structural similarity, SSIM)[16] 是一种测量两幅图像相似性的评价指标, 结合了人类视觉系统 (HVS), 弥补了 PSNR 的一些不足, 测量结果更为准确. SSIM 指数定义如下:

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{8}$$

其中 x 和 y 表示两幅图像的分块, μ_x 和 μ_y 分别表示 x 和 y 的平均值, σ_x^2 和 σ_y^2 分别表示 x 和 y 的方差, σ_{xy} 为协方差, $c_1 = (k_1 L)^2$, $c_2 = (k_2 L)^2$, 这里 $k_1 = 0.01$, $k_2 = 0.03$, L 是灰度级.

本文用 MSSIM[16] 来评价整幅图的质量:

$$\mathrm{MSSIM}(X,Y) = \frac{1}{n}\sum_{i=1}^{n}\mathrm{SSIM}(x_i,y_i), \tag{9}$$

其中 X, Y 分别代表明文图像和密文图像, x_i 和 y_i 表示两幅图对应的第 i 个分块, 是 n 图像分块的个数. MSSIM 越小, 两幅图像差别越大.

3.1.3 亮度相似性 (luminance similarity score, LSS)

LSS[17] 主要是用来计算两幅图像的亮度相似程度指标, 定义如下:

$$\mathrm{LSS}=\frac{1}{n}\sum_{i=1}^{n}f(x_{1i},x_{2i}), \tag{10}$$

$$f(x_{1i},x_{2i})=\begin{cases}1 & (|x_{1i}-x_{2i}|<\beta/2)\\ -\mathrm{round}(|(x_{1i}-x_{2i})/\beta|) & (\text{其他})\end{cases} \tag{11}$$

其中 x_{1i},x_{2i} 分别是两幅图像中第 i 个分块的亮度平均值, 设 $\beta=3$. 如果 LSS 为负则说明两幅图像的亮度极不相似. LSS 越小, 两幅图像差别越大.

3.1.4 边缘相似度 (edge similarity score, ESS)

ESS[17] 是通过计算两幅图像边缘方向的相似程度来确定两幅图像的边缘相似度. 用 1—8 分别表示极坐标系中每 22.5 度分隔的 8 个方向, 0 表示这个分块没有方向. 令 e_{1i} 和 e_{2i} 分别代表两幅图像中第 i 个分块的边缘方向, ESS 可以表示如下:

$$\mathrm{ESS}=\frac{\sum_{i=1}^{n}\omega(e_{1i},e_{2i})}{\sum_{i=1}^{n}c(e_{1i},e_{2i})}, \tag{12}$$

$$\omega(e_{1i},e_{2i})=\begin{cases}0 & (e_{1i}=0\ \text{或}\ e_{2i}=0)\\ |\cos(\phi(e_{1i})-\phi(e_{2i}))| & (\text{其他})\end{cases}, \tag{13}$$

$$c(e_{1i},e_{2i})=\begin{cases}0 & (e_{1i}=e_{2i}=0)\\ 1 & (\text{其他})\end{cases}, \tag{14}$$

其中 $\phi(e)$ 表示方向 e 对应的边缘角度. ESS 的取值范围是从 0 到 1, 0 表示两幅图像的边缘极度不相似; 1 表示边缘信息完全匹配. ESS 越小, 两幅图像差别越大.

表 1 是对 Lena 用本文算法加密后的密文与 AES 加密后的密文的数据对比. 从表 1 可以看出, 这四个指标的数据都非常好, 加密效果优于 AES 算法.

表 2 是 USC-SIPI 图像库中所有的彩色图像的实验数据, 从加密效果部分可以看出各项指标数值很小, 说明加密效果非常好, 该算法能够有效地保护明文信息. 相较于文献 [7, 8, 18], 本文算法具有更

表 1 Lena 密文图像的评价指标数据

Lena	色彩分量	PSNR/dB	MSSIM	LSS	ESS
本文算法	R	6.302	0.003	−24.998	0.023
	G	7.994	0.002	−19.788	0.052
	B	7.931	0.005	−14.487	0.026
	平均值	7.409	0.003	−19.757	0.034
AES	平均值	9.110	0.035	−15.955	0.061

表 2 USC-SIPI 图像库加密结果

文件名	描述	加密效果				压缩效果			
		PSNR /dB	MSSIM	LSS	ESS	PSNR/dB		MSSIM	
						压缩	解密	压缩	解密
4.1.01	Girl	7.730	0.005	−17.784	0.035	34.910	33.822	0.884	0.870
4.1.02	Couple	7.827	0.006	−17.812	0.027	36.272	34.953	0.920	0.901
4.1.03	Girl	10.178	0.029	−10.363	0.031	39.549	38.955	0.928	0.925
4.1.04	Girl	7.407	0.009	−16.334	0.058	37.100	35.262	0.919	0.898
4.1.05	House	8.331	0.015	−17.659	0.082	35.792	35.004	0.873	0.865
4.1.06	Tree	6.441	0.004	−19.992	0.181	32.054	30.869	0.896	0.878
4.1.07	Jelly beans	9.477	0.026	−16.164	0.043	39.105	38.480	0.955	0.953
4.1.08	Jelly beans	8.415	0.010	−16.724	0.076	37.597	36.716	0.954	0.949
4.2.01	Splash	7.208	0.009	−24.782	0.035	37.763	36.848	0.861	0.849
4.2.02	Tiffany	9.499	0.017	−15.435	0.023	35.109	34.588	0.847	0.838
4.2.03	Baboon	6.389	0.002	−14.455	0.133	29.049	27.923	0.886	0.863
4.2.04	Lena	7.409	0.003	−19.757	0.034	35.310	34.555	0.853	0.843
4.2.05	Airplane	8.197	0.010	−16.451	0.061	37.048	35.609	0.915	0.903
4.2.06	Sailboat	6.588	0.004	−21.809	0.096	31.205	30.362	0.828	0.812
4.2.07	Peppers	6.958	0.006	−22.748	0.053	33.399	32.666	0.803	0.791

好的加密效果. 同时对正常解码和加密之后正确解密的结果用 PSNR 和 MSSIM 进行对比, 可以看到正确解密的图像和正常解码的图像质量几乎相同, 因此加密算法对压缩效果没有影响.

3.2 压缩性能

由于直接将空域像素置乱会很大程度上影响压缩效果, 所以本文选择以 8×8 分块为单位进行置乱, 这样可以保证分块内部的关联性, 后续加密过程中, 按照 Zigzag 顺序选择前 n 个 AC 系数进行加密, 会对压缩有一定影响但是影响不大, 在算法能够保证良好加密效果的前提下折中选择 $n=16$. 图 6 给出了不同质量因子下正常压缩图像和加密图像大小对比, 实验图像为 913 kB 的 Lena.bmp, 可以看到对于低质量的图像影响稍大, 这是由于对低质量图像来说 DCT 系数相对较小, 加密过程扩大了 DCT 系数, 造成编码增加; 而对于高质量的图像影响较小. 总体来说加密不会造成太多编码增加, 能获得比原图像小很多的压缩图像.

3.3 格式兼容

本文提出的算法结合了压缩过程并引入选择加密的思想, 在 JPEG 压缩过程中, 只对 DC 系数和部分 AC 系数进行异或操作, 以及将符号比特加密. 在重建图像时, 如果解码器没有获得密钥或者获得错误的密钥, 解码器仍然能够正确解码得到对应的 DCT 系数的值, 并且能够恢复为可视图像, 解码的图像完全无法识别; 如果解码器获得正确的密钥, 那么将进行解密操作, 最后得到正确的恢复图像. 因此本文提出的算法是格式兼容的.

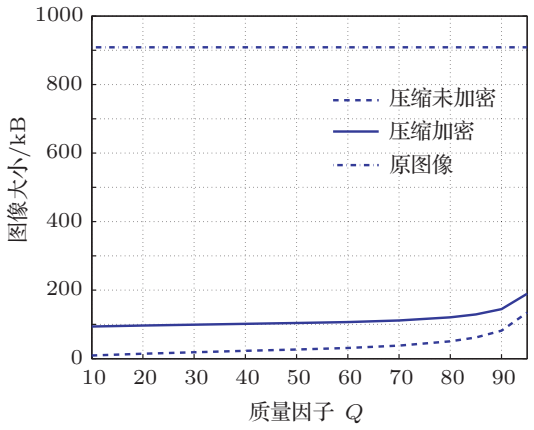

图 6 不同质量因子下压缩图像大小

3.4 加密时间

本文算法中的加密过程主要是置乱和异或操作构成. 置乱的对象是 8×8 块, 相较于对每个像素置乱的方式, 置乱时间可以缩小 1/64; 异或操作运算速度很快, 消耗时间较少. 本文实验代码由 MATLAB 编写, 在未经过优化的情况下, 对于一幅 256×256 的彩色图像加密平均时间约为 1.4 s, 而对于 512×512 的彩色图像加密平均时间约为 5 s. 表 3 是用 USC-SIPI 图像库中彩色图像作为实验对象测试的加密时间和编码时间结果.

表 3 USC-SIPI 图像库加密时间测试结果

文件名	描述	大小/px	加密时间/s	编码时间/s	百分比/%
4.1.01	Girl	256×256	1.54	9.22	16.7
4.1.02	Couple	256×256	1.47	9.33	15.7
4.1.03	Girl	256×256	1.32	7.78	17.0
4.1.04	Girl	256×256	1.57	8.59	18.2
4.1.05	House	256×256	1.22	8.19	14.9
4.1.06	Tree	256×256	1.48	9.51	15.6
4.1.07	Jelly beans	256×256	1.30	7.61	17.0
4.1.08	Jelly beans	256×256	1.28	7.83	16.4
4.2.01	Splash	512×512	4.57	31.20	14.7
4.2.02	Tiffany	512×512	4.78	30.61	15.6
4.2.03	Baboon	512×512	6.13	39.81	15.4
4.2.04	Lena	512×512	5.00	32.16	15.5
4.2.05	Airplane	512×512	5.22	32.66	16.0
4.2.06	Sailboat	512×512	4.99	34.69	14.4
4.2.07	Peppers	512×512	4.83	31.47	15.3

可以看到算法的加密时间平均占编码时间的15.9%, 并且对于不同图像的加密时间百分比也较为稳定, 而文献 [18] 中加密时间比编码时间大了 2 倍左右, 说明本文算法加密时间效率更高加密较为快速.

4 安全性分析

对于选择加密方案, 攻击者可以通过改变密文中特殊的数据值来获取有用的明文信息, 例如引言中提到文献 [1] 将所有 DC 系数置为 128 以及将所有 AC 系数变为正数来获得明文的有效信息, 这种方法可以破解文献 [8] 的算法, 并且对文献 [5, 10] 中算法的安全性有极大威胁; 同时, 文献 [3] 只是利用单纯的 Logistic 映射产生混沌序列, 安全性不够高. 本文提出的算法使用 Henon 映射以及 PWLCM 映射这两个混沌特性较好的映射作为伪随机数发生器, 混沌序列安全性较高; 先将分块进行扩散置乱, 再将每个分块的 DC 系数和重要块中前 16 个 AC 系数加密, 保证攻击者无法简单地利用文献 [1] 的方法来获得有效的明文信息. 重要块标记比特序列经过加密, 随机嵌入在 AC 系数中, 攻击者也无法获取重要块比特序列去得到明文的边缘信息. 这种空域和频域的组合加密方式使得明文和密文间的关系非常复杂, 安全性高. 因此本文提出的算法相较于文献 [3, 5, 8, 10] 的算法有更高的安全性. 为了进一步说明本算法的安全性, 将通过以下几个方面来进一步论证.

4.1 密钥敏感性

本文选择的两个混沌映射都具有良好的敏感性, 初始值微小的改变会导致生成完全不同的混沌序列. 图 7 是一幅对 Lena 用正确密钥解密以及将密钥微小改变为 $K_1 = 0.3600000000000001$ 的解密图像, 可以明显看到用错误密钥解密的结果完全无法识别. 为了更好地说明密钥敏感性, 将 Lena 进行加密之后, 生成 3000 个微小改变的随机密钥 K_1, 其中只有 1 个密钥能正确解密, 用这一组解密密钥生成 3000 幅解密图像, 然后用 PSNR 和 MSSIM 对它们进行评价, 结果如图 8. 可以看到只有正确密钥解

图 7 微小改变密钥解密结果 (a) 用正确密钥解密; (b) 用微小改变的密钥解密

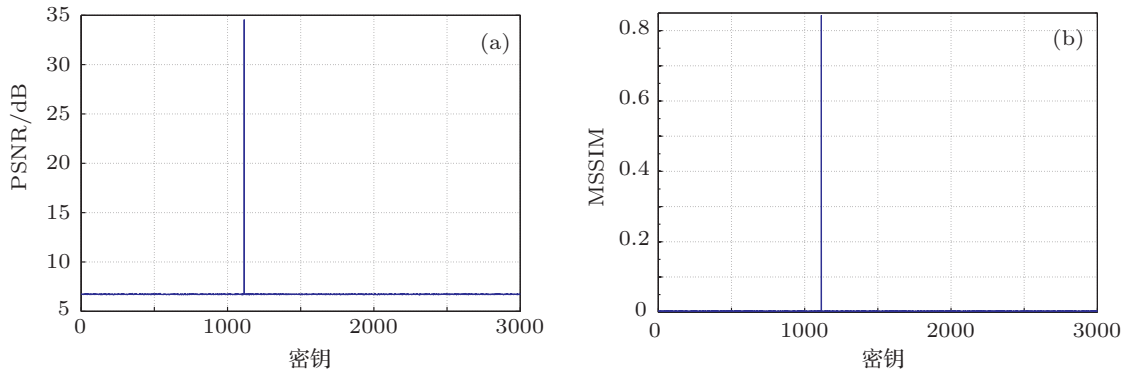

图 8 用 3000 个不同的 K_1 解密 Lena 的结果 (a) PSNR; (b) MSSIM

密的图像有最高的 PSNR 和 MSSIM, 其余的错误密钥解密图像 PSNR 和 MSSIM 都非常低. 并且对 K_2, K_3, K_4, K_5 进行同样微小改变之后解密, 可以得到同样的结果. 说明本文算法具有很高的密钥敏感性.

4.2 密钥空间

本文算法中密钥 K_1, K_2, K_3, K_4, K_5 均为十进制浮点数, 经测试, 在小数点后 16 位微小变化能产生完全不同的混沌序列, 而小数点后 17 位微小变化则不会产生改变, 因此每个密钥的可变化极限为 10^{-16}. K_1 是混沌 1 初值 x_0, 取值范围 (0, 1), 密钥空间为 1×10^{16}; K_2 是混沌 1 的参数 μ, 取值范围 (0, 0.5), 密钥空间为 0.5×10^{16}; K_3 是混沌 2 的初值 x_0, 取值范围 $(-1.5, 1.5)$, 密钥空间为 3×10^{16}; K_4 是混沌 3 的初值 x_0, 取值范围 (0, 1), 密钥空间为 1×10^{16}, K_5 是混沌 4 的初值 x_0, 取值范围 (0, 1), 密钥空间为 1×10^{16}. 因此本算法总的密钥空间为 $10^{16\times3}\times0.5\times10^{16}\times3\times10^{16}\approx2^{267}$, 对比 AES 算法密钥长度 128 或 256 位, 3-DES 算法密钥长度为 112 或 168 位, 本文算法密钥长度足够安全, 并且相较于文献 [2, 6—8] 具有更大的密钥空间, 能保证密钥无法被穷举破解.

4.3 平均攻击

如果将加密认为是一种图像噪声, 去噪算法或者图像恢复的手段则可以削弱或者去掉噪声. 图像平均 [19] 被认为是一种有效的手段, 它通过将同一图像的多幅噪声图像平均来得到去噪图像. 将多幅 Lena 的错误解密图像视作样本噪声图像, 通过将各个彩色分量对应位置像素进行平均的方式来得到去噪图像 [20], 实验结果列于表 3 中. 可以看到随着样本图像数量的增加, PSNR 和 MSSIM 有轻微的增加, 但是不足以获得有效的明文信息. 而且实际上随着样本图像的增加, 最后去噪图像会变得较为均匀, 却无法识别任何明文信息. 因此本文算法不能以图像去噪的方式恢复任何有用信息.

表 4 对若干 Lena 密文图像进行平均的结果

图像数量	2	4	8	16	32	64	128	256	512	1024
PSNR	8.017	8.790	9.257	9.496	9.630	9.695	9.729	9.741	9.751	9.754
MSSIM	0.004	0.005	0.005	0.006	0.006	0.006	0.006	0.006	0.006	0.006

5 结 论

本文算法主要结合空域置乱操作和边缘检测的方式, 先加密所有 DC 系数, 再选择包含细节信息较多的重要块进行重点加密, 保证信息完全不被泄露; 同时将分块重要性信息嵌入 AC 系数中, 使密钥传输更合理. 算法结合了选择加密思想并将加密操作嵌入到 JPEG 彩色图像压缩编码过程中, 使得本算法具有良好的格式兼容性. 从实验结果可以看出算法的加密效果好, 并且解决了现有算法中存在的一些安全问题, 安全性得到提升. 在未来进一步的工作中将优化加密算法来进一步减少加密对压缩效果的影响.

[1] Wu C P, Kuo C C J 2005 *IEEE Trans. Multimedia* **7** 828

[2] Yuen C H, Wong K W 2011 *Appl. Soft. Comput.* **11** 5092

[3] Ge X, Liu F, Lu B, Wang W, Chen J 2010 *2nd IEEE Int. Conf. Information Management and Engineering (ICIME)* Chengdu, China, April 16–18, 2010 267

[4] 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 Kwok-Wo, 张伟, 韦鹏程 2012 物理学报 **61** 040505]

[5] Lian S 2009 *Chaos Soliton. Fract.* **40** 2509

[6] Yuen C H, Wong K W 2012 *Chin. Phys.* B **21** 010502

[7] Lian S, Sun J, Wang Z 2004 *8th IEEE Int. Conf. Information Visualisation* London, UK, July 14–16, 2004 217

[8] Deng J X, Deng H T 2013 *Chin. Phys.* B **22** 094202

[9] Xu P, Zhao J, Wang D A 2011 *3rd IEEE Int. Conf. Communication Software and Networks (ICCSN)* Xi'an, China, May 27–29, 2011 376

[10] Krikor L, Baba S, Arif T, Shaaban Z 2009 *Eur. J. Sci. Res.* **32** 47

[11] Luo Y, Du M, Liu D 2012 *5th IEEE Int. Work. Chaos-Fractals Theories and Applications (IWCFTA)* Dalian, China, October 18–21, 2012 191

[12] 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]

[13] Bhatnagar G, Wu Q M J 2012 *IEEE Trans. Instrum. Meas.* **61** 876

[14] Canny J 1986 *IEEE Trans. Pattern Anal. Mach. Intell.* **8** 679

[15] The USC-SIPI image database http: //sipi. usc. edu/database/

[16] Wang Z, Bovik A C, Sheikh H R, Simoncelli E P 2004 *IEEE Trans. Image Process* **13** 600

[17] Mao Y, Wu M 2004 *IEEE Int. Conf. Image Processing (ICIP'04)* Singapore, Singapore, October 24–27, 2004 569

[18] Wen C C, Wang Q, Huang F M, Liu X H, Chen X Z 2012 *J. Comput. -Aided Design Comput. Graph.* **24** 500 (in Chinese) [文昌辞, 王沁, 黄付敏, 刘向宏, 陈新中 2012 计算机辅助设计与图形学学报 **24** 500]

[19] Gonzalez R C, Woods R E (translated by Ruan Q Q, Ruan Y Z) 2007 *Digital Image Processing,* (2nd Ed.) (Beijing: Publishing House of Electronics Industry) (in Chinese) p88 [R C 冈萨雷斯, R E 伍德著 (阮秋琦, 阮宇智译) 2007 数字图像处理 (第二版) (北京: 电子工业出版社) 第 88 页]

[20] Zhao L, Liao X F, Xiang T, Xiao D 2010 *Acta Phys. Sin.* **59** 1507 (in Chinese) [赵亮, 廖晓峰, 向涛, 肖迪 2010 物理学报 **59** 1507]

A joint compression and encryption scheme for color JPEG image*

Xiao Di† Xie Yi-Jun

(*Key Laboratory of Dependable Service Computing in Cyber Physical Society, Ministry of Education, College of Computer Science, Chongqing University, Chongqing* 400044, *China*)

(Received 24 July 2013; revised manuscript received 17 September 2013)

Abstract

To research the encryption algorithm for color joint photographic expert group (JPEG) image, by comprehensively choosing the selective encryption and the joint compression and encryption, an encryption algorithm combining with spatial domain and frequent domain for color JPEG image is proposed. The 8×8 blocks are first diffused on spatial domain, then the edge-detection method is utilized to find out the significant blocks containing abundant details. After encrypting all the direct current coefficients, part of alternating current (AC) coefficients in significant blocks are chosen to be encrypted. Finally the information about marking the significance of the blocks is embedded into AC coefficients for transmission. Theoretical analyses and experimental results show that the proposed algorithm is compatible with JPEG format. The cipher image has good visual quality and uniform color distribution. The algorithm possesses huge key space, strong key-sensitivity and good security.

Keywords: color JPEG image, edge-detection, selective encryption, joint compression and encryption

PACS: 05.45.Gg **DOI:** 10.7498/aps.62.240508

* Project supported by the Natural Science Foundation of Chongqing Science and Technology Commission, China (Grant No. CSTC2011JJJQ40001).

† Corresponding author. E-mail: dixiao@cqu.edu.cn