计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2021.20200172
58(6):1340-1350,2021

基于预测误差自适应编码的图像加密可逆数据隐藏

杨尧林[1] 和红杰[1] 陈 帆[1] 原长琦[2]
[1](西南交通大学信息科学与技术学院 成都 611756)
[2](北京电子技术应用研究所 北京 100091)
(ylyangwr@foxmail.com)

Reversible Data Hiding of Image Encryption Based on Prediction Error Adaptive Coding

Yang Yaolin[1], He Hongjie[1], Chen Fan[1], and Yuan Changqi[2]
[1](*College of Information Science and Technology, Southwest Jiaotong University, Chengdu* 611756)
[2](*Beijing Institute of Electronics Technology and Application, Beijing* 100091)

Abstract For the security problem of existing schemes in the image encryption, and the problem of low compression due to poor coding, this paper proposes a reversible data hiding algorithm of image encryption based on prediction error adaptive coding. In the image encryption stage, an image encryption algorithm based on error maintenance is designed. First, block scrambling and pixel modulation encryption are performed on 3×3 image blocks, and then non-center pixels are grouped and scrambled according to the central pixel value of the image block. In the data embedding stage, adaptive coding is based on the prediction error distribution of the image, after marking and classifying the pixels with the coding table, and the coding table and additional data are hidden together in the encrypted image to generate a marked encrypted image. The experimental results show that group scrambling operation in the encryption phase increases the number of eigenvalue difference blocks between the original image and the encrypted image, makes it difficult to determine the correspondence between the image blocks in the image before and after encryption, improves the security of the encrypted image, and keeps the overall prediction error distribution of the image. Compared with state-of-the-art algorithms, the average embedding rate can be improved by more than 0.49 bpp, the additional data can be extracted losslessly and the original image can be restored.

Key words reversible data hiding; error maintenance encryption; group scrambling; adaptive coding; privacy protection

摘 要 针对现有算法中加密图像存在安全隐患,及选用编码不佳导致图像压缩率较低的问题,提出了一种基于预测误差自适应编码的图像加密可逆数据隐藏算法.图像加密阶段,设计了一种基于误差维持的图像加密算法,首先对3×3的图像块做块间置乱和像素调制加密,然后根据图像块中心像素值将非中心像素分组置乱.数据嵌入阶段,根据图像自身预测误差分布自适应编码,使用编码表对像素进行标记分类后,将编码表与附加数据共同隐藏在加密图像中生成携密加密图像.实验结果表明:加密阶段

收稿日期:2020-03-17;**修回日期**:2020-07-24
基金项目:国家自然科学基金项目(61872303,U1936113);四川省科技厅科技创新人才计划项目(2018RZ0143)
This work was supported by the National Natural Science Foundation of China (61872303, U1936113) and the Science and Technology Innovation Talents Program of Sichuan Science and Technology Department (2018RZ0143).
通信作者:陈帆(fchen@ swjtu.edu.cn)

分组置乱操作,使原始图像与加密图像中特征值差异块数增多,难以确定加密前后图像中各图像块间的对应关系,提高了加密图像的安全性,且图像整体的预测误差分布保持不变;相较于现有算法,平均嵌入率提高 0.49 bpp 以上,且能无损提取附加数据、恢复原始图像.

关键词 可逆数据隐藏;误差维持加密;分组置乱;自适应编码;隐私保护

中图法分类号 TP391

随着计算机的发展应用,更多的用户选择把数据上传至云端存储,云存储使数据的所有权和管理权分离,使得云存储中数据安全以及个人隐私保护引起了人们的关注,加密图像可逆数据隐藏(reversible data hiding in encrypted image, RDH-EI)可为云存储中图像数据的安全提供技术支持[1-2].与传统的明文域可逆数据隐藏不同,RDH-EI 首先对图像进行加密,在加密图像中实现可逆的数据隐藏.其中,图像加密有效避免了原始图像内容的泄露.同时,密文图像中附加数据的提取可用于图像真实性认证,来源追踪及隐私保护等多种应用场景[3-4].

现有 RDH-EI 可分为 2 类:加密前预留空间(reserving room before encryption, RRBE)[5-9]和加密后腾出空间(vacating room after encryption, VRAE)[10-18].RRBE-RDHEI 可利用明文图像像素间的相关性提高嵌入容量,但增加了图像拥有者的操作难度,不仅需要对原始图像进行加密,还要在加密前执行预处理操作,这对于普通用户来说是难以实现的.而在 VRAE-RDHEI 中,图像拥有者只需要执行图像加密即可,减小了图像拥有者的技术需求.

在已有的 VRAE-RDHEI 算法中,根据嵌入方法的不同可分为 3 类:1)采用低位翻转[10-12],通过翻转加密图像像素的最低有效位(least significant bit, LSB)嵌入附加数据.该类算法操作简单,可获得较高的直接解密图像质量,但数据提取与图像恢复 2 个步骤不可分离,且图像恢复阶段不能完全可逆.2)采用传统 RDH 嵌入附加数据,其中直方图移位[13-15]应用较广泛,通过寻找像素值或误差值的峰值点与零点值,将像素值进行小幅度修改实现信息的嵌入.该类算法能够实现完全可逆,但嵌入容量相对较小.3)采用编码无损压缩,使用编码表对图像进行压缩,从而腾出空间用于嵌入附加数据.该类算法嵌入容量较高,得到多数学者的关注.Yi 等人[16]提出一种参数加密域可逆数据隐藏(parametric reversible data hiding in encrypted images, PRDHEI)算法,设计了 2 种嵌入编码策略,在嵌入过程中选取嵌入率较高的编码表进行标记压缩.在图像完全恢复的情况下,平均嵌入率达 1.19 bpp.为进一步提高嵌入率,Yi 等人[17]修改了嵌入过程中使用的编码表.在二叉树逻辑结构的基础上实用化,提出了参数二叉树标记(parametric binary tree labeling, PBTL)的方法,将加密后图像的像素根据预测误差及选取的参数标记为可嵌入像素和不可嵌入像素 2 种,标记后压缩的空间可用来嵌入秘密信息.该算法的平均嵌入率可达 1.68 bpp,相较于文献[16]的嵌入率提高 0.49 bpp.上述 2 种方案对可嵌入像素预测误差的编码长度是相等的,但对分布不均匀的预测误差,变长编码要比定长编码具有更高的压缩率.在 Fu 等人[18]提出的自适应编码策略算法中,使用预先设定的哈夫曼编码表对加密后图像块的最高有效位(most significant bit, MSB)分类标记,平均嵌入率可达 1.81 bpp.该方案中编码表涵盖了一个块中最多存在 4 种不同 MSBs 的情况,但对纹理图像,包含 4 种 MSBs 以上的块数是较多的,将其全部作为不可嵌入块将会导致嵌入率降低.综上所述,现有 VRAE-RDHEI 文献中使用的编码表大多是预先设定的,未结合图像自身的特征,因此根据图像自身像素值特征选取适合的编码策略是提高嵌入容量的一种有效解决方案.

提高嵌入容量是已有 RDH-EI 算法研究的主要内容,同时,加密算法的安全性也逐渐被研究者们关注.上述算法中,文献[7-12]采用了位异或加密方法,加密后的熵值较高,密文图像类似随机噪声,但加密前后像素位置并未发生变化,不能抵抗 Khelifi[19]提出的唯密文攻击.为抵抗这类唯密文攻击算法,多位研究者提出块置乱及相应改进算法[13-18].在文献[13]中,采用块内置乱与块间置乱,该方法中像素值并未发生改变,不能抵抗 Li 等人[20]提出的已知明文攻击.在文献[16-17]中,块间置乱与像素调制后虽然像素的位置与像素值均发生变化,但块内像素的相对差值依旧会保留.在文献[15]中,块间置乱与块内异或后由于块内像素进行异或运算的二进制值序列相同,所以块内像素的部分相关性将会保留,且在块间置乱前后,加密图像块与原始图像块是一一对应的关系.上述加密算法中,原始图像与加密图像

块之间存在的关联性与块内像素的相关性成为攻击者破解加密图像的依据.为提高加密算法的安全性,打乱置乱操作保留的一一对应关系是需要研究的关键问题.

为提高算法嵌入容量的同时兼顾加密算法的安全性,本文提出一种基于预测误差自适应编码的图像加密可逆数据隐藏算法.在加密阶段,采用误差维持加密算法,包括块间置乱、像素调制以及分组置乱3个步骤.与未加入分组置乱的加密算法相比,增加了加密前后图像的特征值差异块数,提高了加密图像的安全性,同时未改变加密图像的整体预测误差分布.在嵌入阶段,根据加密图像的预测误差分布自适应编码(adaptive coding, AC),生成哈夫曼编码表后对像素进行标记压缩,能实现较大的嵌入容量.且哈夫曼编码具有唯一标识性,能无损提取附加数据与恢复原始图像,实现完全可逆.

1 基于预测误差自适应编码的 RDH-EI

本文算法主要包括3部分:基于误差维持的图像加密;基于自适应编码的信息嵌入;数据提取与图像恢复.1)基于误差维持的图像加密.图像拥有者利用加密密钥 K_{enc} 对原始图像 I_{ori} 执行块间置乱与像素调制,再将非中心像素分组置乱生成加密图像 I_{enc}.2)基于自适应编码的信息嵌入.数据隐藏者根据加密图像 I_{enc} 的预测误差分布选取范围生成哈夫曼编码表,对图像进行标记后嵌入辅助数据与加密的附加数据生成携密加密图像 I_{mark}.3)数据提取与图像恢复.图像接收者根据持有的密钥可分别获得原始的附加数据或图像.下面对算法按上述3个部分进行详细描述.

1.1 基于误差维持的图像加密

在误差维持加密中,图像拥有者首先将图像分为非重叠块,对所有块执行块间置乱与像素调制;然后进行分组置乱,通过图像块的中心像素值将对应的非中心像素分组后,根据加密密钥将每组的像素进行置乱.具体操作描述如下:

1) 块间置乱与像素调制

Step1. 图像分块.将大小为 $A\times B$ 的原始图像 I_{ori} 分为 m 个大小为 3×3 的非重叠块 $I_{\text{ori}}^{(k)}$ ($k=1,2,\cdots,m$),其中 $m=\lfloor A/3\rfloor\times\lfloor B/3\rfloor$.

Step2. 中间图像生成.根据加密密钥 K_{enc},采用文献[16]中的方案执行块间置乱与像素调制.若图像未被整除,则对边缘区域的像素进行异或加密,生成中间图像 I_{int},其中包含非重叠块 $I_{\text{int}}^{(k)}$ ($k=1,2,\cdots,m$).

2) 分组置乱

Step1. 各组像素数量统计.将每个 $I_{\text{int}}^{(k)}$ 的中心像素组成大小为 $\lfloor A/3\rfloor\times\lfloor B/3\rfloor$ 的图像 I_{ic}.根据直方图统计图像 I_{ic} 中像素值为 p($0\leqslant p\leqslant255$)的像素数量 n_p,可得在中间图像 I_{int} 中,中心像素为 p 所对应的非中心像素数量为 $n_{p\text{non}}=8\times n_p$.

Step2. 非中心像素分组置乱.将图像 I_{int} 中心像素值为 p 的图像块中非中心像素分为1组,生成分组序列 $C_p=\{C_p^1,C_p^2,\cdots,C_p^{n_{p\text{non}}}\}$.基于加密密钥 K_{enc} 产生 $1\sim n_{p\text{non}}$ 互不相同的伪随机自然数序列 $T_p=\{T_p^1,T_p^2,\cdots,T_p^{n_{p\text{non}}}\}$.使用序列 T_p 置乱对应的分组序列 C_p,得置乱后的分组序列 D_p 为

$$D_p(i)=C_p(T_p^i),\ i=1,2,\cdots,n_{p\text{non}}. \tag{1}$$

Step3. 加密图像生成.将置乱后的分组序列 D_p 依次重新排列到图像 I_{int} 中心像素值为 p 的图像块中,生成加密图像 I_{enc}.

本文误差维持加密算法中的分组置乱操作,将中间图像中心像素值相同块对应的非中心像素随机置乱.一方面,将块间置乱保留的原始图像块与加密图像块一一对应关系变为多对多的关系,攻击者难以确定原始图像块与加密图像块之间的关联,提高了加密图像的安全性.另一方面,同一个非中心像素在分组置乱前后所在图像块的中心像素值一致,即非中心像素与中心像素的预测误差值保持不变.

为更直观地描述加密过程,以图像大小为 9×9,分块大小为 3×3 为例描述加密操作,如图1所示.图1(a)为原始图像,可分为4个像素块,灰色部分为中心像素;图1(b)为块置乱图像,由原始图像按置乱序列{3,4,2,1}执行块间置乱产生;图1(c)为中间图像,通过调制序列{1,2,7,6}对块置乱图像执行像素调制生成.最后根据中心像素的不同可分为3组,分别为中心像素值为101的非中心像素序列{101,102,104,104,105,106,104,109};中心像素值为105的序列{104,104,102,106,102,106,105,105,108,108,110,115,108,116,105,106};中心像素值为110的序列{110,109,108,108,115,108,109,106}.分别按置乱序列{2,5,8,6,3,1,4,7},{5,16,14,3,9,10,4,15,12,1,6,8,11,2,13},{5,7,3,6,4,2,1,8}执行置乱并重新排列后生成如图1(d)所示的加密图像.

1.2 基于自适应编码的信息嵌入

当数据隐藏者接收到加密图像后,首先根据预测误差分布选取范围对像素进行分类;然后根据

各类像素概率生成并保存哈夫曼编码表;最后使用编码表对像素进行标记与附加数据嵌入.具体操作描述为:

1) 像素分类

Step1. 临界概率值确定.根据同层概率相同的哈夫曼树中编码概率与编码长度的关系确定临界概率值.假设临界概率表示为 $p_{\lim}$,编码长度为 l,则两者的关系为

$$p_{\lim}=\left(\frac{1}{2}\right)^{l}. \tag{2}$$

对于 8 b 深度的灰度图像像素值,哈夫曼编码长度不能超过 8 b,则编码临界概率至少满足 $p_{\lim}=0.00390625$.

102	102	104	103	102	101
109	99	102	101	103	108
110	99	100	101	102	99
100	101	103	102	102	100
103	100	104	104	103	100
105	103	108	104	103	103

(a) 原始图像

100	101	103	102	102	100
103	100	104	104	103	100
105	103	108	104	103	103
103	102	101	102	102	104
101	103	108	109	99	102
101	102	99	110	99	100

(b) 块置乱图像

101	102	104	104	104	102
104	101	105	106	105	102
106	104	109	106	105	105
110	109	108	108	108	110
108	110	115	115	105	108
108	109	106	116	105	106

(c) 中间图像

102	105	109	102	106	116
106	101	104	102	105	105
101	104	104	108	108	102
115	109	108	105	115	104
108	110	108	106	105	105
109	110	106	110	104	108

(d) 加密图像

□ 非中心像素 ■ 中心像素

Fig. 1 Schematic of encryption process

图 1 加密过程示意图

Step2. 预测误差计算.首先将加密图像 I_{enc} 分为 m 个大小为 3×3 的非重叠块 $I_{\text{enc}}^{(k)}(k=1,2,\cdots,m)$.然后将第 k 个块的中心像素定义为参考像素,记为 $E_{\text{ref}}^{(k)}$,其余的 8 个像素定义为非参考像素,记为 $E_i^{(k)}$ $(i=1,2,\cdots,8)$.则第 k 个块中第 i 个非参考像素与参考像素的预测误差 $e_i^{(k)}$ 为

$$e_i^{(k)}=E_i^{(k)}-E_{\text{ref}}^{(k)}. \tag{3}$$

Step3. 非参考像素分类.通过直方图统计整幅加密图像预测误差值为 $e(e\in[-255,255])$ 的数量 n_e,对应概率为 p_e.由临界概率值 $p_{\lim}$ 确定小于 0 的预测误差下限 α 与大于 0 的预测误差上限 β.根据 α 和 β 将非参考像素分为 2 类,若第 k 个块中第 i 个非参考像素的预测误差 $e_i^{(k)}$ 满足:

$$\alpha\leqslant e_i^{(k)}\leqslant\beta, \tag{4}$$

则该像素 $E_i^{(k)}$ 属于可嵌入像素;否则,属于不可嵌入像素.

2) 编码表生成与保存

Step1. 编码表的生成.将所有不可嵌入像素归为一类,则其概率为

$$p_{e\text{non}}=\sum_{e=-255}^{\alpha-1}p_e+\sum_{e=\beta+1}^{255}p_e. \tag{5}$$

可嵌入像素根据预测误差不同可分 $\lambda=\beta-\alpha+1$ 类,其概率为 $p_e(e=\alpha,\alpha+1,\cdots,\beta-1,\beta)$,则非参考像素共分为 $\lambda+1$ 类.根据 $\lambda+1$ 类像素的概率生成哈夫曼编码表,其中,不可嵌入像素对应的编码为 f_0,λ 类可嵌入像素对应的编码为 $f_i(i=1,2,\cdots,\lambda)$.

若编码表中存在编码的长度大于 8 b,将预测误差下限 α 加 1,预测误差上限 β 减 1,对非参考像素重新分类后再次执行本步骤.

Step2. 哈夫曼编码表存储结构构造.分别用 $\lceil\text{lb }256\rceil=8$ b 二进制序列存储 $|\alpha|$ 与 $|\beta|$,用于表示连续的可嵌入像素预测误差范围.然后,统计编码 $f_i(i=0,1,\cdots,\lambda)$ 的长度 d_i,用 $\lceil\text{lb }8\rceil=3$ b 表示.最终构造的哈夫曼编码表存储结构如图 2 所示:

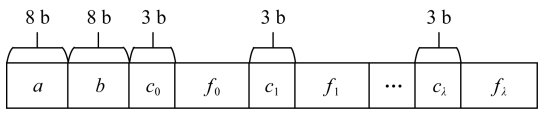


Fig. 2 Storage form of Huffman coding table

图 2 哈夫曼编码表存储结构

其中,a,b,$c_i(i=0,1,\cdots,\lambda)$ 分别为 $|\alpha|$,$|\beta|$,d_i 对应的二进制序列.哈夫曼编码存储长度为 η.

Step3. 编码表保存.将哈夫曼编码表通过位替换存储在前 n_{ref} 个块的参考像素中,当 η 不足 8 的倍数时,在末尾补充长度为 $8\times n_{\text{ref}}-\eta$ 的比特‘0’序列.被替换参考像素的原始比特组合为 w.

3)像素标记与附加数据嵌入

Step1. 不可嵌入像素标记.通过位替换将不可嵌入像素的前 d_0 位 MSBs 替换为 f_0,剩余的 $8-d_0$ 位 LSBs 保持不变.同时将被替换的原始 MSBs 组合的比特流记为 v,其长度为 ξ.

Step2. 可嵌入像素标记与附加数据嵌入.用长度为 $d_i(i=1,2,\cdots,\lambda)$ 的编码 f_i 替换可嵌入像素的 MSBs,剩余的 $8-d_i$ 位 LSBs 嵌入总隐藏数据 φ,生成携密加密图像 I_{mark}.

在上述总隐藏数据 φ 中,除加密的附加数据 ϕ 外,还包括一部分图像的原始数据作为辅助数据用于图像恢复.其中,ϕ 为原始附加数据 ρ 通过数据隐藏密钥 K_{hid} 异或加密后生成的比特流.辅助数据包括 2 部分:保存编码表时被替换的前 n_{ref} 个块的参考像素原始比特流 w;不可嵌入像素的原始 d_0 位 MSBs 组成的比特流 v.将总隐藏数据长度定义为 ζ,则本算法的嵌入率 $rate$ 为

$$rate=\frac{\zeta-8\times n_{\text{ref}}-\xi}{A\times B}. \tag{6}$$

综上所述,在基于自适应编码的嵌入过程中,通过加密图像整体的预测误差分布特性自适应地生成哈夫曼编码表,对像素进行标记压缩腾出空间用于数据嵌入.其中,哈夫曼编码的唯一标识性成为数据提取与图像恢复阶段对携密加密图像中的非参考像素进行分类的依据.

1.3 数据提取和图像恢复

图像接收者得到携密加密图像 I_{mark} 后,根据持有密钥的不同,其权限也不同.拥有数据隐藏密钥 K_{hid},可提取原始的附加数据 ρ;拥有加密密钥 K_{enc},可恢复原始图像 I_{ori}.

1) 附加数据提取

Step1. 分类界限提取.将携密加密图像 I_{mark} 分为 m 个大小为 3×3 的非重叠块 $I_{\text{mark}}^{(k)}(k=1,2,\cdots,m)$,并从图像块 $I_{\text{mark}}^{(1)}$ 与 $I_{\text{mark}}^{(2)}$ 的参考像素中分别提取 8 b 恢复预测误差下限 α 与预测误差上限 β,确定可嵌入像素类数 $\lambda=\beta-\alpha+1$.

Step2. 编码表重建.从 $I_{\text{mark}}^{(3)}$ 与其随后块的参考像素中依次提取 $\lambda+1$ 个编码的编码长度 $d_i(i=0,1,\cdots,\lambda)$ 与编码 f_i,重建哈夫曼编码表,同时统计存储编码表被替代的参考像素个数 n_{ref}.

Step3. 附加数据的提取.根据重建的哈夫曼编码表与非参考像素的标记位,将非参考像素分为可嵌入像素与不可嵌入像素,对不可嵌入像素,统计其数量 $n_{e\text{non}}$,对可嵌入像素,提取除标记位外的 $8-d_i$ 位 LSBs,得到总隐藏数据 φ.可知,总嵌入数据的前 $8\times n_{\text{ref}}$ 位为保存编码表时被替换的参考像素原始比特流 w,中间 $n_{e\text{non}}\times d_0$ 位为不可嵌入像素的前 d_0 位 MSBs 组成的比特流 v,剩余位为加密的附加数据 ϕ.最后用数据隐藏密钥 K_{hid} 将 ϕ 进行异或解密可得原始的附加数据 ρ.

2) 图像恢复

Step1. 加密像素值的恢复.首先,与附加数据提取一致,提取 w 与 v.然后,用 w 恢复前 n_{ref} 个块的原始参考像素.非参考像素中,对不可嵌入像素,用 v 恢复原始像素的前 d_0 位 MSBs;对可嵌入像素,由编码表中编码与预测误差的一一对应关系可得出预测误差值,进而恢复加密图像 I_{enc} 的原始像素值.

Step2. 原始图像的恢复.加密过程具有可逆性,使用加密密钥 K_{enc} 对加密图像 I_{enc} 依次执行逆分组置乱、逆像素调制、逆块间置乱即可恢复原始图像 I_{ori}.

综上所述,数据提取与图像恢复阶段能够无损重建哈夫曼编码表,进而根据所持密钥无损地提取附加数据或恢复原始图像.

2 实验结果与分析

下面通过实验来验证分析本文算法的性能,主要从 4 个方面:1)误差维持加密算法分析;2)嵌入容量分析;3)可逆性分析;4)运行时间分析.在实验过程中选取如图 3 所示的大小为 512×512 的 8 幅灰度测试图像,分别为 Lena,Jetplane,Barbara,Peppers,Boat,Lake,Crowd,Baboon.同时使用包含 1 338 幅的 UCID[21] 与包含 10 000 幅的 BOSSbase[22] 这 2 个图像库进行实验分析,其中,UCID 中所有图像均转变为 512×512 的灰度图像.

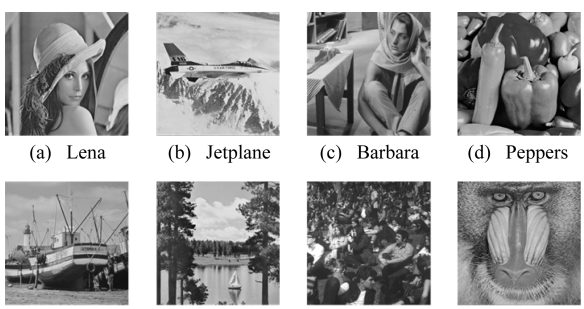


Fig. 3 Test image

图 3 测试图像

2.1 误差维持加密算法分析

误差维持加密包括块间置乱、像素调制和分组置乱 3 部分,其中,分组置乱操作是根据所在块中心像素值的不同,将非中心像素分组后置乱.该操作能改变非中心像素所在的块,打乱块间置乱操作保留的原始图像与加密图像中各图像块间的对应关系,提高加密算法的安全性.且分组置乱前后非中心像素所在块的中心像素值相等,故整体的预测误差分布不变.下面从这 2 方面对加密算法进行分析测试.

1) 整体预测误差分布

为证明分组置乱是否会造成整体预测误差分布

发生变化,对分组置乱的原理进行分析.在分组置乱的过程中,所有相同中心像素值所对应的非中心像素分为一组进行置乱,此时即使改变非中心像素的位置,它们所对应的中心像素值依旧保持不变.而预测误差值是将非中心像素值减去中心像素值,因此只要中心像素值未发生变化,预测误差值均保持不变.以图 1(c)(d)为例,在图 1(c)中,中心像素值为 105 对应的非中心像素为{104,104,102,106,102,106,105,105,108,108,110,115,108,116,105,106},预测误差为{−1,−1,−3,1,−3,1,0,0,3,3,5,10,3,11,0,1}.而在图 1(d)中,中心像素值为 105 对应的非中心像素预测误差为{−3,1,11,−3,0,3,3,−3,0,10,−1,1,0,5,−1,3},可以看出,图 1(c)中间图像与图 1(d)加密图像的预测误差分布一致.

以 Lena 图像为例,统计有无分组置乱加密图像(中间图像与加密图像)的预测误差分布,结果如图 4 所示.可以看出,两者的预测误差分布相同,说明加入分组置乱后并未改变整体的预测误差分布,即不会影响图像的嵌入容量.

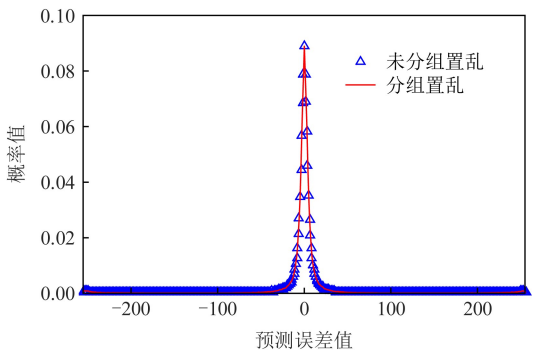


Fig. 4 Prediction error distribution with or without group scrambling

图 4 有无分组置乱的预测误差分布

2) 加密算法安全性分析

为评价加密图像的安全性,定义原始图像与加密图像的特征值差异块数作为衡量指标,其中,图像块的特征值是该块所有的非中心像素与中心像素差值绝对值的总和.当原始图像与加密图像的特征值差异块数越多时,表明加密图像的安全性越高.计算指标的具体操作为:

1) 计算图像块特征值,定义第 k 个块的特征值为 8 个非中心像素的预测误差绝对值总和 $e_{\mathrm{sum}}^{(k)}$:

$$e_{\mathrm{sum}}^{(k)}=\sum_{i=1}^{8}\left|e_i^{(k)}\right|, \tag{7}$$

其中,$e_i^{(k)}$ 为第 k 个块中第 i 个非参考像素的预测误差.

2) 通过直方图统计原始图像、加密图像中图像块特征值为 pe 的数量 N_{ori}^{pe},N_{enc}^{pe}.由于非参考像素的预测误差值 $e\in[-255,255]$,得 pe 的取值范围为 $pe\in[0,255\times 8]$.

3) 统计原始图像与加密图像之间的特征值差异块数 dif:

$$dif=\frac{1}{2}\times\sum_{pe=0}^{255\times 8}\left|N_{\mathrm{ori}}^{pe}-N_{\mathrm{enc}}^{pe}\right|. \tag{8}$$

为验证本文加密算法的安全性,以加入分组置乱的加密图像与未加入分组置乱的中间图像进行对比.

首先,以图 1 为例,图 1(a)中原始图像各块的特征值分别为{36,17,27,10},图 1(c)中间图像各块的特征值分别为{27,10,17,36},图 1(d)加密图像各块的特征值分别为{27,27,17,21}.此时原始图像与中间图像特征值差异块数为 0,块间置乱序列能够被唯一确定,而原始图像与加入了分组置乱加密图像的特征值差异数为 2,存在一半的块无法确定对应关系,因此加入分组置乱能提高加密图像的安全性.

然后,以 8 幅测试图像为例,统计原始图像与中间图像的特征差异块数 dif_{oi},原始图像与加密图像的特征差异块数 dif_{oe},结果如表 1 所示.可以看出,dif_{oe} 相较于 dif_{oi} 增加了 3～4 倍,即加入了分组置乱后的加密图像与原始图像的特征值差异块数较多,难以确定原始与加密图像中图像块的对应关系,加密图像的安全性得到提高.

Table 1 Comparison of Eigenvalue Difference Blocks

表 1 特征值差异块数对比

测试图像	dif_{oi}	dif_{oe}
Lena	1 915	9 721
Jetplane	1 823	10 486
Barbara	3 095	11 028
Peppers	1 927	7 612
Boat	2 279	11 618
Lake	2 721	9 306
Crowd	2 324	10 829
Baboon	4 267	8 762

下面从密钥空间的角度对本文加密算法的安全性进行分析.假设图像大小为 $A\times B$ 且分块大小为 3×3,则块间置乱的密钥空间 ψ_1 为

$$\psi_1=\left(\left\lfloor\frac{A}{3}\right\rfloor\times\left\lfloor\frac{B}{3}\right\rfloor\right)!, \tag{9}$$

像素调制的密钥空间 ψ_2 为

$$\psi_2 = 256^{\left(\lfloor \frac{A}{3} \rfloor \times \lfloor \frac{B}{3} \rfloor\right)}, \tag{10}$$

像素调制后图像的像素值近似均匀分布，故分组置乱的密钥空间 ψ_3 为

$$\psi_3 = \left[\left(\left\lfloor \frac{A}{3} \right\rfloor \times \left\lfloor \frac{B}{3} \right\rfloor \times \frac{8}{256}\right)!\right]^{256}. \tag{11}$$

本文加密算法的密钥空间为 $\psi = \psi_1 \times \psi_2 \times \psi_3$. 当图像大小为 512×512 且分块为 3×3 时，可能生成 $28900! \times 256^{28900} \times (903!)^{256}$ 种不同的加密图像. 在这种情况下，若没有加密密钥，攻击者很难将加密图像进行恢复.

综上所述，误差维持加密算法中，在加入分组置乱后未改变图像整体预测误差分布，但增加了加密图像与原始图像的特征值差异块数，提高了加密图像的安全性.

2.2 嵌入容量分析

由 RDH-EI 算法易知，总嵌入容量与压缩率正相关，即压缩率越高，总嵌入容量越高，嵌入率也就越高.以 8 幅测试图像为例，将文献[17]的 PBTL 在最大嵌入容量时的相关数据与本文进行对比，结果如表 2 所示.其中，ζ 为总隐藏数据的长度，即总嵌入容量；ξ 为不可嵌入像素原始 MSBs 比特流的长度，η 为哈夫曼编码表存储结构的长度，ξ 与 η 之和为辅助数据的总长度；γ 为附加数据的长度，即净嵌入容量；根据式(6)计算得到图像的嵌入率 *rate*.分析 Lena 图像的测试数据可知，本文算法相较于 PBTL，ζ 提高 99.89 Kb，ξ 提高 7 713 b，η 提高 286 b，γ 提高 92.07 Kb，嵌入率 *rate* 提高 0.36 bpp.统计 8 幅测试图像的平均值，本文算法的 ζ 提高 70.27 Kb，ξ 降低 14.71 Kb，η 提高 364 b，γ 提高 84.62 Kb，嵌入率 *rate* 提高 0.33 bpp.

Table 2 Comparison of the Embedding Capacity and Auxiliary Data Between the Algorithm of Ref [17] and Our Algorithm

表 2 文献[17]与本文算法的嵌入容量与辅助数据对比

测试图像	算法	ζ/b	ξ/b	η/b	γ/b	*rate*/bpp
Lena	PBTL[17]	594 273	66 216	8	528 049	2.014
	本文	696 557	73 929	294	622 332	2.374
Jetplane	PBTL[17]	691 152	116 822	8	574 322	2.191
	本文	769 565	87 939	261	681 362	2.599
Barbara	PBTL[17]	475 851	145 164	8	330 679	1.261
	本文	495 909	96 850	431	398 627	1.521
Peppers	PBTL[17]	614 472	79 125	8	535 339	2.042
	本文	683 056	69 390	292	613 370	2.340
Boat	PBTL[17]	548 967	96 420	8	452 539	1.726
	本文	653 467	96 120	358	556 987	2.125
Lake	PBTL[17]	517 191	117 604	8	399 579	1.524
	本文	571 701	102 609	385	468 700	1.788
Crowd	PBTL[17]	549 765	95 888	8	453 869	1.731
	本文	685 990	93 489	368	592 133	2.259
Baboon	PBTL[17]	325 986	136 412	8	189 566	0.723
	本文	337 073	112 858	590	223 623	0.853

为进一步验证本算法的嵌入容量优势，选取 3 篇同类文献(VRAE-RDHEI)进行统计对比，包括 Yi 等人[16]的 PRDHEI 算法、Yi 等人[17]的 PBTL 算法和 Fu 等人[18]的自适应编码算法.其中，文献[16]采用算法完全可逆时的嵌入容量；文献[17]选取测试图像的最大嵌入容量；文献[18]设定分块大小为 4×4、MSB 位数 $H=5$、阈值 $T=4$ 时的嵌入容量. 以 8 幅测试图像为例，对比文献与本文算法最大嵌入容量如表 3 所示.分析表 3 数据可知，本文的嵌入容量均高于现有文献.结合表 2 分析，对较平滑的 Jetplane 图像，预测误差分布相对集中，压缩率较高，嵌入率相较 3 篇对比文献分别提高了 1.324 bpp，0.408 bpp，0.406 bpp；对于较纹理的 Baboon 图像，预测误差分布较均匀，压缩率较低，相较对比文献提升较少，分别为 0.853 bpp，0.130 bpp，0.391 bpp.对 8 幅测试图像的嵌入率计算平均值可得，本文算法

的平均嵌入率相较对比文献分别提高了 1.262 bpp,0.330 bpp,0.364 bpp.

Table 3 Comparison of Maximum Embedding Rate

表 3 最大嵌入率对比 bpp

测试图像	Ref [16]	Ref [17]	Ref [18]	本文
Lena	0.701	2.014	2.018	2.374
Jetplane	1.275	2.191	2.193	2.599
Barbara	0.347	1.261	1.355	1.521
Peppers	0.544	2.042	1.977	2.340
Boat	0.767	1.726	1.810	2.125
Lake	0.301	1.524	1.442	1.788
Crowd	1.107	1.731	1.685	2.259
Baboon	—	0.723	0.462	0.853
Average	0.720	1.652	1.618	1.982

注:“—”表示无法嵌入信息.

为更好地说明本文算法的嵌入率高于现有算法,对 UCID 的 1 338 幅图像与 BOSSbase 的 10 000 幅图像分别测试,计算每个图像库的平均值作为算法的平均嵌入率,得到本文与文献[16-18]在 2 个图像库的平均嵌入率如表 4 所示.为方便测试,文献[16]设定分块大小为 3×3,$\alpha=5$,$\beta=2$ 时的嵌入容量作为最大嵌入容量.通过对比表 4 数据可知,对 UCID 图像库,本文的平均嵌入率相较于对比文献分别提高 1.115 bpp,0.626 bpp,0.492 bpp.对 BOSSbase 图像库,分别提高 1.251 bpp,0.881 bpp,0.662 bpp.

Table 4 Comparison of Average Embedding Rate of Database

表 4 图像库平均嵌入率对比 bpp

数据库	Ref [16]	Ref [17]	Ref [18]	本文
UCID	1.191	1.680	1.814	2.306
BOSSbase	1.584	1.954	2.173	2.835

为可视化本文与对比文献在嵌入容量上的对比情况,从 UCID 与 BOSSbase 数据库中分别选取前 200 幅图像进行测试,结果如图 5 所示.可以看出,本文算法所得到的嵌入率要高于 3 篇对比文献.在 UCID 的 200 幅图像中,本文算法的平均嵌入率相较于文献[16-18]分别提高 1.24 bpp,0.45 bpp,0.46 bpp;在 BOSSbase 的 200 幅图像中,本文算法的平均嵌入率分别提高 1.24 bpp,0.71 bpp,0.58 bpp.

为对比分析本文 AC 方案与现有腾挪空间方法的性能,本文新增加 2 篇文献[7-8]RRBE-RDHEI 对比分析.文献[7-8]均采用中值边缘检测(median edge detection, MED)方法预测,腾挪空间方法分

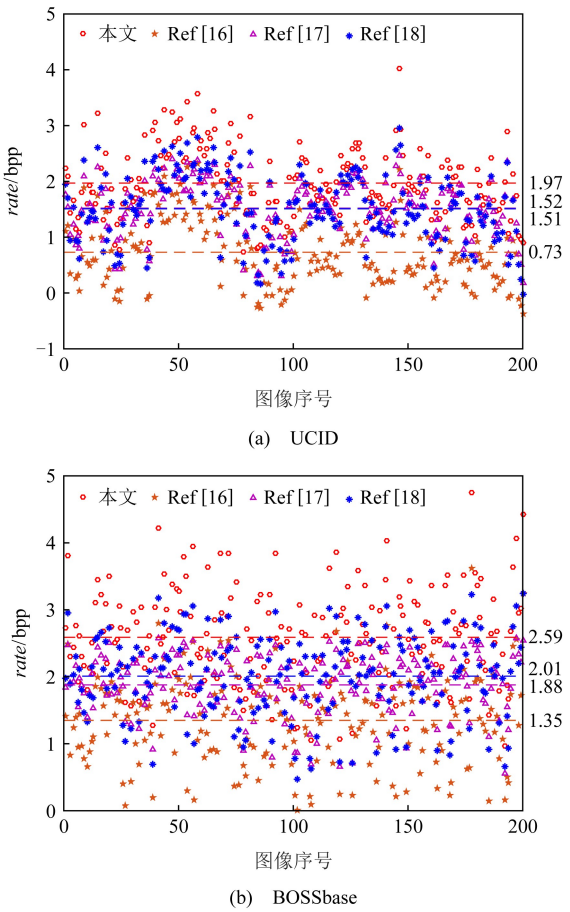


Fig. 5 Comparison of the embedding rate between our algorithm and three state-of-the-art algorithms

图 5 本文算法与 3 篇现有算法的嵌入率对比

别为提高的参数二叉树标记的可逆数据隐藏方案(improved reversible data hiding scheme in encrypted images using parametric binary tree labeling, IPBTL)和哈夫曼编码标记(Huffman coding labeling, HVLCL).在实验中,为公平起见,本文采用相同的 MED 预测方法得到预测图像,并按照文献[7]对算法的加密阶段进行调整.调整后算法中生成含标记加密图像的步骤描述为:1)生成预测误差分布图.将原始图像第 1 行与第 1 列作为参考像素进行 MED 预测生成预测误差分布图.2)图像加密.对原始图像进行异或加密生成加密图像.3)像素分组.结合本文的临界概率值与预测误差分布图,将非参考像素分为可嵌入像素与不可嵌入像素.4)像素标记.由本文的 AC 方案生成哈夫曼编码表,并构造出哈夫曼编码的存储结构自左至右保存到加密图像的第 1 行参考像素中.结合预测误差分布图对加密图像进行标记,将加密图像原始参考像素比特流及不可嵌入像素被替换的原始比特流嵌入可嵌入像素压缩出的空间,生成

含标记的加密图像.最终,剩余空间可被数据隐藏者用于嵌入附加数据.

然后,分别使用 IPBTL[7]、HVLCL[8]和本文的调整算法得到相应的最大嵌入容量,其中,IPBTL[7]设定 $\alpha=5$,$\beta=3.8$ 幅测试图像的对比结果如表 5 所示.以 Lena 图像为例,本文 AC 相较于 IPBTL 的总嵌入容量提高 167.34 Kb,辅助数据提高 17.24 Kb,净嵌入容量提高 150.10 Kb,嵌入率提高 0.587 bpp;相较于 HVLCL 的总嵌入容量降低 543.00 Kb,辅助数据降低 718.00 Kb,净嵌入容量提高 175.01 Kb,嵌入率提高 0.683 bpp.同理,计算 8 幅图像对比结果求取平均值可得,IPBTL 的平均嵌入率为 2.256 bpp,HVLCL 的平均嵌入率为 2.284 bpp,本文 AC 方案的平均嵌入率为 2.842 bpp,相较于 IPBTL 和 HVLCL 分别提高 0.586 bpp 与 0.558 bpp.

Table 5 RRBE-RDHEI Scheme Comparison of Test Image

表 5 测试图像的 RRBE-RDHEI 方案对比

测试图像	算法	总嵌入容量/b	辅助数据/b	净嵌入容量/b	嵌入率/bpp
Lena	IPBTL[7]	742 494	40 877	701 617	2.676
	HVLCL[8]	1 469 869	793 764	676 105	2.579
	AC	913 845	58 528	855 317	3.263
Jetplane	IPBTL[7]	745 968	37 403	708 565	2.703
	HVLCL[8]	1 587 492	792 417	795 075	3.033
	AC	1 017 307	67 592	949 715	3.623
Barbara	IPBTL[7]	635 919	147 452	488 467	1.863
	HVLCL[8]	1 313 541	840 006	473 535	1.806
	AC	712 374	113 772	598 602	2.283
Peppers	IPBTL[7]	734 568	48 803	685 765	2.616
	HVLCL[8]	1 386 368	783 654	602 714	2.299
	AC	835 452	50 800	784 652	2.993
Boat	IPBTL[7]	731 070	52 301	678 769	2.589
	HVLCL[8]	1 470 649	794 852	675 797	2.578
	AC	895 112	60 940	834 172	3.182
Lake	IPBTL[7]	676 374	106 997	569 377	2.172
	HVLCL[8]	1 302 121	785 980	516 141	1.969
	AC	722 935	68 000	654 935	2.498
Crowd	IPBTL[7]	725 766	57 605	668 161	2.549
	HVLCL[8]	1 560 448	789 132	771 316	2.942
	AC	966 510	72 452	894 058	3.411
Baboon	IPBTL[7]	506 763	276 608	230 155	0.878
	HVLCL[8]	1 074 525	794 591	279 934	1.068
	AC	496 510	107 357	389 153	1.485

为进一步说明本文 AC 方案优于现有的 IPBTL[7]与 HVLCL[8],以 UCID 与 BOSSbase 两个图像库为例进行实验,结果如表 6 所示.可以看出,在相同预测方法的基础上,本文的 AC 方案具有最高的嵌入容量,相较于 IPBTL[7]提高 1.11 bpp 以上,相较于 HVLCL[8]提高 0.51 bpp 以上.

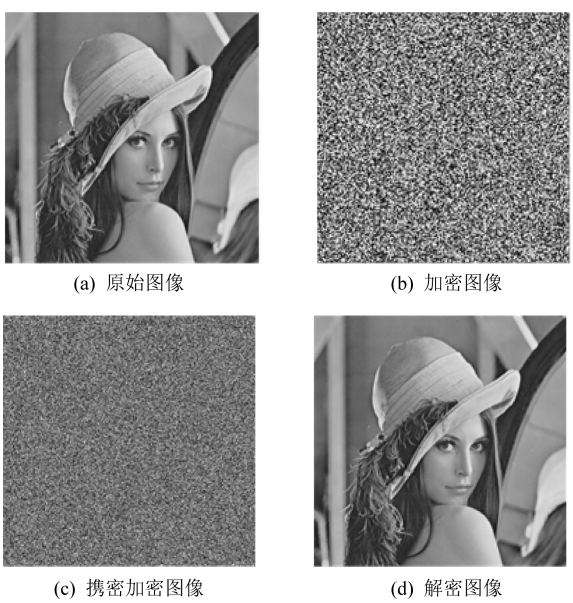
Table 6 RRBE-RDHEI Scheme Comparison of Database

表 6 图像库的 RRBE-RDHEI 方案对比 bpp

数据库	IPBTL[7]	HVLCL[8]	AC
UCID	2.478	3.072	3.586
BOSSbase	2.567	3.361	3.921

2.3 可逆性分析

本文算法中,采用哈夫曼编码对像素进行标记,基于哈夫曼编码的唯一标识性,能够实现原始图像的可逆恢复.图 6 显示了 Lena 图像在本算法中的仿真结果.其中图 6(a)为原始的 Lena 图像,图 6(b)为加密图像,图 6(c)为携密加密图像,图 6(d)为恢复图像.使用峰值信噪比(peak signal-to-noise ratio, *PSNR*)与结构相似性(structural similarity, *SSIM*)作为评测算法的可逆性的指标,则恢复图像的 *PSNR* 为∞,*SSIM* 为 1,本文算法实现完全可逆.



(a) 原始图像 (b) 加密图像

(c) 携密加密图像 (d) 解密图像

Fig. 6 Simulation results of Lena

图 6 Lena 图像仿真结果

为进一步验证算法的可逆性,测试 2 个图像库的相关指标,如表 7 所示.可以看出,本算法在最大、最小嵌入率的图像中,以及图像库的平均情况下,恢复图像的 *PSNR* 均趋近于无穷,*SSIM* 均为 1,说明

可以实现完全可逆.且本算法在 2 个图像库中的最高嵌入率可达 5 bpp 以上,平均嵌入率可达 2.3 bpp 以上.综上,本算法在实现高嵌入容量的同时能完全可逆恢复原始图像.

Table 7 Test Results of Database

表 7 图像库测试结果

图像库	嵌入情况	*PSNR*/dB	*SSIM*	*rate*/bpp
UCID	最高	∞	1	5.025
	最低	∞	1	0.003 9
	平均	∞	1	2.306
BOSSbase	最高	∞	1	5.993
	最低	∞	1	0.434
	平均	∞	1	2.835

2.4 运行时间分析

RDH-EI 算法主要包括 4 个阶段:图像加密、数据嵌入、数据提取和图像恢复,其中,图像所有者用户主要执行图像加密算法,因此加密性能直接影响用户的体验.下面对本文加密算法的运行时间分析,并选择 5 篇文献作对比:文献[7]的异或加密;文献[13]的块间置乱与块内置乱加密;文献[17]的块间置乱与像素调制加密;文献[15]的块间置乱与块内异或加密;文献[18]的块间置乱、块内置乱与块内异或加密.

时间复杂度上分析说明:当图像大小为 $A\times B$ 且分块大小为 $s\times s$ 时,异或加密、块内异或、块内置乱与像素调制对所有像素执行了一次操作,时间复杂度为 $O(A\times B)$,块间置乱的操作以块为单位进行,时间复杂度为 $O(A\times B/s^2)$,本文分组置乱操作中除参考像素外的其余像素均进行置乱,时间复杂度为 $(s^2-1)/s^2\times O(A\times B)$.

对各种加密算法进行实验对比,实验环境为:Windows 10 操作系统(企业版 2016)、MATLAB 2015a;硬件配置为 Intel® Core™ i5-6200U CPU@ 2.30 GHz 2.40 GHz,4.00 GB 内存(3.89 GB 可用),64 位操作系统的笔记本电脑.

实验测试图像:大小为 256×256,512×512,1024×1024 的图像;分块大小为 2×2 及 3×3.运行时间统计 10 次求取平均值,结果如表 8 所示.可以看出,本文加密算法在提高算法安全性前提下,图像大小为 512×512 且分块大小为 3×3 时的运行时间为 1.28 s,在可接受范围内.

Table 8 Comparison of Running Time of Encryption Algorithms

表 8 加密算法运行时间对比 s

尺寸		Ref [7]	Ref [13]	Ref [17]	Ref [15]	Ref [18]	本文
图像	图像块						
256×256	2×2	0.21	0.43	0.25	0.25	0.56	0.56
	3×3	0.21	0.32	0.11	0.11	0.38	0.32
512×512	2×2	0.87	1.76	1.00	1.00	2.26	2.26
	3×3	0.88	1.26	0.45	0.45	1.49	1.28
1024×1024	2×2	3.61	7.35	4.29	4.26	9.52	10.11
	3×3	3.53	5.10	1.86	1.87	6.06	5.53

3 结 论

本文提出一种基于预测误差自适应编码的图像加密可逆数据隐藏算法.误差维持图像加密算法打乱了块间置乱操作保留的对应关系,提高了加密图像的安全性.同时保留整体预测误差的分布,不会降低嵌入算法的性能.数据可逆嵌入阶段,利用图像的预测误差分布生成哈夫曼编码表用于自身标记压缩,相较于现有预先设定编码表或定长编码的方案具有更高的压缩性能,实现了更高的嵌入容量,以 UCID 图像库为例,平均嵌入率可达 2.306 bpp.后续工作将从更高效的临界概率方案设计、图像加密算法安全性分析、算法时间复杂度优化等方面展开.

参 考 文 献

[1] Shi Yunqing, Li Xiaolong, Zhang Xinpeng, et al. Reversible data hiding: Advances in the past two decades [J]. IEEE Access, 2016, 4: 3210-3237

[2] Yan Shu, Chen Fan, He Hongjie. Reversibledata hiding in encrypted image based on neighborhood prediction using XOR-permutation encryption [J]. Journal of Computer Research and Development, 2018, 55(6): 1211-1221 (in Chinese)
(鄢舒, 陈帆, 和红杰. 异或-置乱框架下邻域预测加密域可逆信息隐藏[J]. 计算机研究与发展, 2018, 55(6): 1211-1221)

[3] Qian Zhenxing, Zhou Hang, Zhang Xinpeng, et al. Separable reversible data hiding in encrypted JPEG bitstreams [J]. IEEE Transactions on Dependable and Secure Computing, 2018, 15(6): 1055-1067

[4] Ke Yan, Zhang Minqing, Su Tingting. Anovel multiple bits reversible data hiding in encrypted domain based on R-LWE [J]. Journal of Computer Research and Development, 2016, 53(10): 2307-2322 (in Chinese)

(柯彦, 张敏情, 苏婷婷. 基于 R-LWE 的密文域多比特可逆信息隐藏算法[J]. 计算机研究与发展, 2016, 53(10): 2307-2322)

[5] Ma Kede, Zhang Weiming, Zhao Xianfeng, et al. Reversible data hiding in encrypted images by reserving room before encryption [J]. IEEE Transactions on Information Forensics and Security, 2013, 8(3): 553-562

[6] Yuan Yuan, He Hongjie, Chen Fan. Reduction of the redundancy of adjacent bit planes for reversible data hiding in encrypted images [J]. Journal of Image and Graphics, 2019, 24(1): 13-22 (in Chinese)
(袁源, 和红杰, 陈帆. 减少相邻位平面间冗余度的加密图像可逆信息隐藏[J]. 中国图象图形学报, 2019, 24(1): 13-22)

[7] Wu Youqing, Xiang Youzhi, Guo Yutang, et al. An improved reversible data hiding in encrypted images using parametric binary tree labeling [J]. IEEE Transactions on Multimedia, 2020, 22(8): 1929-1938

[8] Yin Zhaoxia, Xiang Youzhi, Zhang Xinpeng. Reversible data hiding in encrypted images based on multi-MSB prediction and Huffman coding [J]. IEEE Transactions on Multimedia, 2020, 22(4): 874-884

[9] Qiu Yingqiang, Ying Qichao, Lin Xiaodan, et al. Reversible data hiding in encrypted images with dual data embedding [J]. IEEE Access, 2020, 8: 23209-23220

[10] Zhang Xinpeng. Reversible data hiding in encrypted image [J]. IEEE Signal Processing Letters, 2011, 18(4): 255-258

[11] Hong Wien, Chen Tungshou, Wu Hanyan. An improved reversible data hiding in encrypted images using side match [J]. IEEE Signal Processing Letters, 2012, 19(4): 199-202

[12] Li Ming, Xiao Di, Peng Zhongxian, et al. A modified reversible data hiding in encrypted images using random diffusion and accurate prediction [J]. ETRI Journal, 2014, 36(2): 325-328

[13] Yin Zhaoxia, Luo Bin, Hong Wien. Separable and error-free reversible data hiding in encrypted image with high payload [J]. The Scientific World Journal, 2014, 2014: No.604876

[14] Yu Chunqiang, Ye Chenmei, Zhang Xianquan, et al. Separable reversible data hiding in encrypted image based on two-dimensional permutation and exploiting modification direction [J]. Mathematics, 2019, 7(10): No.976

[15] Huang Fangjun, Huang Jiwu, Shi Yunqing. New framework for reversible data hiding in encrypted domain [J]. IEEE Transactions on Information Forensics and Security, 2016, 11(12): 2777-2789

[16] Yi Shuang, Zhou Yicong. Parametric reversible data hiding in encrypted images using adaptive bit-level data embedding and checkerboard based prediction [J]. Signal Processing, 2018, 150: 171-182

[17] Yi Shuang, Zhou Yicong. Separable and reversible data hiding in encrypted images using parametric binary tree labeling [J]. IEEE Transactions on Multimedia, 2019, 21(1): 51-64

[18] Fu Yujie, Kong Ping, Yao Heng, et al. Effective reversible data hiding in encrypted image with adaptive encoding strategy [J]. Information Sciences, 2019, 494: 21-36

[19] Khelifi F. On the security of a stream cipher in reversible data hiding schemes operating in the encrypted domain [J]. Signal Processing, 2018, 143: 336-345

[20] Li Shujun, Li Chengqing, Chen Guanrong, et al. A general quantitative cryptanalysis of permutation-only multimedia ciphers against plaintext attacks [J]. Signal Processing, 2008, 23(3): 212-223

[21] Schaefer G, Stich M. UCID: An uncompressed color image database [J]. Proceedings of SPIE, 2004, 5307: 472-480

[22] Bas P, Filler T, Pevný T. "Break our steganographic system": The ins and outs of organizing BOSS [C] //Proc of the 13th Int Conf on Information Hiding. Berlin: Springer, 2011: 59-70

Yang Yaolin, born in 1996. Master candidate. His main research interests include image processing and reversible data hiding in encrypted domain.
杨尧林,1996 年生.硕士研究生.主要研究方向为图像处理和加密域可逆数据隐藏.

He Hongjie, born in 1971. PhD, professor. Member of CCF. Her main research interests include digital image processing and information security.
和红杰,1971 年生.博士,教授,CCF 会员.主要研究方向为数字图像处理和信息安全.

Chen Fan, born in 1971. PhD. His main research interests include multimedia security and digital watermarking.
陈 帆,1971 年生.博士.主要研究方向为多媒体安全和数字水印.

Yuan Changqi, born in 1984. Master, assistant professor. His main research interest is crypto theory.
原长琦,1984 年生.硕士,助理研究员.主要研究方向为加密理论.