

# 利用预测误差直方图平移的可逆数据隐藏技术

李建伟 胡永健 陈开英

(华南理工大学自动化科学与工程学院, 广州 510641)

**摘要** 根据图像差值直方图的特点,提出了一种在图像差值中进行差值扩展嵌入的可逆数据隐藏技术。为了减少数据嵌入所带来的图像质量退化,提出了几种策略,包括选择交错平移直方图以控制差值移动的个数,利用直方图零点对直方图进行分段以减小在创造嵌入空间时直方图平移的位移,以及一种与直方图平移技术相适应并随负载变化压缩尺寸的溢出位置图。实验表明,本算法比现有文献中的典型算法(例如, Thodi 等人的算法和 Tian 的算法)具有更好的整体性能。在中、低嵌入率下,本文算法性能更加突出。

**关键词** 可逆数据隐藏 无损水印 直方图平移 JBIG 压缩

中图法分类号: TP391.4 文献标识码: A 文章编号: 1006-8961(2009)06-1088-08

## Reversible Data Hiding Technique Using Difference Histogram Shifting

LI Jian-wei, HU Yong-jian, CHEN Kai-ying

(College of Automation Science and Engineering, South China University of Technology, Guangzhou, 510641)

**Abstract** Based on the characteristics of difference histograms, we propose a reversible data hiding algorithm using difference expansion. In order to decrease the embedding distortion, we present three schemes, including interleavingly shifting the outer regions of the histogram, dividing the outer regions into segments by using zero points, and a payload-dependent overflow location map. This first scheme is used to control the number of selected differences; the second one is used to decrease the shifting distance of the region movement; the last one is used to generate a compact compressed overflow location map. The experimental results verify that, compared with other typical reversible algorithms in the literature (e.g., Thodi et al's and Tian's algorithms), the proposed algorithm has better overall performance. In particular, our algorithm has more advantages at low- and middle-embedding rates.

**Keywords** reversible data hiding, lossless watermark, histogram shifting, JBIG compression

## 1 引言

可逆数据隐藏(又称无损数据隐藏)是近几年数据隐藏和水印处理领域一个新的研究分支,由于其适用于特殊的应用场合,在信息安全领域越来越被重视。它的主要思想是,向宿主信号嵌入一个数串作为负载或水印信号,得到嵌入数据或加水印后的宿主信号。在接收端,用户一方面可从嵌入数据或加水印的宿主信号中无损地提取所嵌入的数据或

水印,另一方面还可以无损地恢复出原始宿主信号。这种可逆的数据隐藏技术在许多对图像质量要求高的应用场合有非常好的应用前景,例如,在军事图像和医学图像诊断等<sup>[1]</sup>。

文献中已经出现不少无损/可逆数据隐藏(水印)算法<sup>[1-7]</sup>。Tian 首先提出了利用差值扩展(difference expansion)的方法实现可逆的数据隐藏<sup>[2]</sup>,该算法有较高的嵌入容量,且容易扩展。后来不少算法<sup>[4-7]</sup>都是对 Tian 的方法的改进和扩展。Thodi 等人<sup>[7]</sup>利用差值扩展技术和直方图平移技术

基金项目:国家自然科学基金项目(60572140,60772115);广东省自然科学基金项目(04020004)

收稿日期:2007-08-16;改回日期:2007-10-17

第一作者简介:李建伟(1982~),男,华南理工大学模式识别与智能系统专业硕士研究生。主要研究方向为信息隐藏技术、图像处理与模式识别。E-mail:lijianwei012@gmail.com

相结合,对 Tian 的算法进行了改进,大大提高了输出图像的质量。然而,Thodi 等人的算法仍然存在不足,例如,在纹理图像中,其嵌入数据后的图像质量不是很高。

受 Thodi 等人算法的启发,本文利用差值扩展和直方图平移提出一种新的基于预测误差直方图的可逆数据隐藏技术。首先根据预测误差直方图不完全对称的特点,以交错平移直方图外区域的方式控制要改变的差值的个数。为了进一步降低图像质量的退化,利用直方图零点将直方图外区域分段,以分段平移的方式减少平移的位移。此外,还提出一种依负载而变的溢出位置图,只需要很少的数据就可记录可能发生溢出的位置,从而间接地确定嵌入位置。本文算法大大改善了嵌入数据后的输出图像的质量,在同等的嵌入率(尤其是中、低嵌入率)下,本文方法通常比现有文献中最新的可逆数据嵌入算法—Thodi 等人的算法<sup>[7]</sup>具有更高的图像质量。本文算法的性能远远超过了 Tian 所提出的可逆数据嵌入的经典算法<sup>[2]</sup>。

## 2 基于预测误差直方图平移的算法

首先定义本文将会用到的部分符号。 $x, \hat{x}, x'$  分别表示原像素值、预测值和修改后的像素值;预测误差  $p_e = x - \hat{x}$ ;  $p'_e$  表示嵌入后的预测误差值; $i$  表示待嵌入的二进制比特。在选择可扩展的差值时,还将用到两个属于正整数类型的差值选择阈值  $T_n$  和  $T_p$ ,其中  $T_n$  用于左半轴,也称为左阈值; $T_p$  用于右半轴,也称为右阈值。 $P$  表示负载数据; $Q$  表示头文件; $M$  表示 JBIG 压缩后的溢出位置图; $\eta(\cdot)$  表示求数据长度的函数(单位:比特); $count$  记录被选中进行扩展嵌入的差值个数。由于直方图零点在本算法中扮演重要角色,给出下面的定义:

**定义 1:** 在差值直方图中,如果差值  $p_e$  和其所发生的次数(occurrence)  $f(p_e)$  存在这样的关系:当  $p_e \neq 0$  时,  $f(p_e) = 0$ ,称此时的  $p_e$  为直方图上的一个零点。

零点一般位于直方图的尾部端点以外,然而,有时也会出现在直方图的尾部端点以内,甚至直方图的中部。本文要利用的就是这类零点。通常,做扩展嵌入用的区域被称为内区域,而正、负半轴上  $T_p + 1, -T_n$  分别和它们最远离原点的非零端点之间的区域被称作外区域(如图 1 所示)。本

文将利用直方图尾部端点以内的零点把外区域分段。

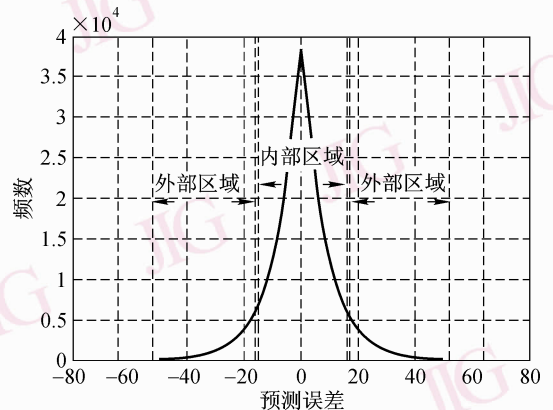


图 1 一般图像的预测误差直方图

Fig. 1 Histogram of predicted errors for general images

### 2.1 算法原理

本文算法的基础是在图像差值中通过差值扩展进行数据嵌入,选用差值的目的是将数据隐藏于高频分量,这样不会引起明显的视觉畸变。为了进一步降低嵌入误差,优先选择差值绝对值接近于 0(也就是在横轴上原点附近)的差值。根据图像编码的知识,预测误差直方图的分布一般服从拉普拉斯分布,因此,嵌入位置是从直方图的对称轴处开始,在其两侧由小到大地进行差值的选取。很显然,数值为零的差值( $p_e = 0$ )是首选。

本文算法的核心是选择可扩展嵌入的差值的位置,选择的过程是:首先根据差值直方图,确定上述首选差值  $p_e = 0$  的差值是向左扩展嵌入,变成  $-1/0$ ,还是向右扩展嵌入,变成  $0/1$ 。即要确定直方图左边区域先移动,还是右边区域先移动。若朝直方图左边(以下简称负向)移动,其相匹配的嵌入公式为  $p'_e = 2p_e - i$ ;若朝直方图右边(以下简称正向)移动,则相匹配的嵌入公式为  $p'_e = 2p_e + i$ 。为了叙述方便,本文假设首次移动为负向。正向的操作和负向对称。一旦确定平移方向,就可根据负载的大小在直方图两侧以渐进的方式交错地扩大图 1 中的内部区域,并通过外区域的直方图平移,创造“空闲的”空间。隐藏数据的扩展嵌入将发生在位于内部区域的差值上,而嵌入后的内部区域由于差值的绝对值变大会占据外区域平移所创造“空闲的”空间。为了使得数据抽取后图像可以无损地恢复,必须保证内外区域不重叠,即避免嵌入后的差值和图像原来的差值重叠。而为了减少形变,利用直方图端点

以内的零点将直方图分段,这样外区域的平移每次只涉及相关的段,而其他段可以保持不变。注意,内区域差值的扩展嵌入和外区域差值的平移都会改变图像的像素值,从而引起形变。这就是总是设法尽可能少地改动内外区域的原因。

用图 2 解释上面内区域的扩展和外区域的平移。即将嵌入的区间(内区域)是  $[-T_n + 1, T_p]$ 。嵌入后的区间变成  $[-2T_n + 1, 2T_p]$ 。而外区域也只改变了相关的段,并没有改动除开内区域的全部直方图。这一点将在后面解释。

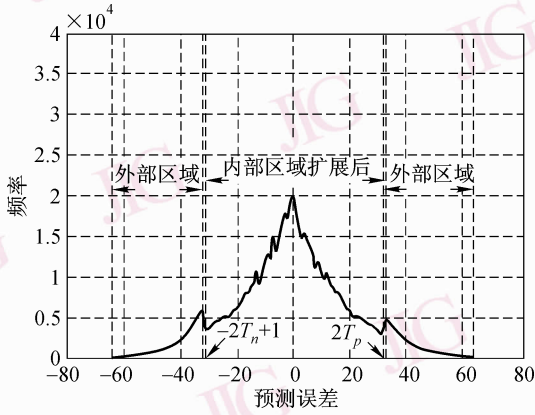


图 2 内、外区域的变化和直方图分段的作用

Fig. 2 Change of inner and outer regions and effect of histogram subsection

### 2.1.1 确定直方图首次移动的方向

对于给定负载, Thodi 等人的算法<sup>[7]</sup>是同时把外区域向横轴的两端平移来获取空闲空间,在直方图平坦的情况下,需要移动的差值数量非常大,这会严重影响图像质量,这也是本文算法在纹理图像中性能下降的主要原因。本文算法用交替平移外区域来克服这个缺点,只有在空闲空间不够的时候才平移外区域,而且一次移动一侧,当恰好满足负载的要求时立即停止移动。因此本文算法必须确定首先移动直方图哪一侧的外部区域,即嵌入时  $p_e = 0$  的差值向哪一侧扩展。值得指出的是,首次扩展方向将确定要使用的嵌入公式。一旦选定嵌入公式,则后面所有的嵌入就依此公式进行。

确定首次移动或称扩展方向的做法是:在直方图两侧分别选择两个固定的区间,例如  $[-10, -1]$  和  $[1, 10]$ ,比较这两个区间中的可嵌入差值点的个数,即两个区间中全体  $f(p_e)$  的和,哪个区间的可嵌入点个数少,首次嵌入时 0 就向哪一边扩展。由于发生差值溢出的位置通常不多,所以在确定首次移

动方向时可以暂不考虑溢出位置的影响,直接比较两个区间的差值个数。从经验上看,在这样大小的区间中取值可以反映直方图某一侧各个差值出现频数的总体情况。

### 2.1.2 根据负载的大小以渐进的方式交错地扩大内部区域

主要介绍如何确定待嵌入差值的位置。设所确定的首次扩展方向为负向,则算法开始创造空闲空间(寻找待嵌入差值的位置)的循环,第一轮时  $T_n = 1, T_p = 0$ 。在直方图负半轴(内区域以外)选择最靠近原点的一个零点  $-Z_{n1}$ ,则此时的内区域为  $[0, 0]$ ,外区域为  $[-Z_{n1} + 1, -T_n]$ 。为了创造空闲空间,必须把外区域  $[-Z_{n1} + 1, -T_n]$  左移一个单位,变成  $[-Z_{n1}, -T_n - 1]$ ,这样内区域才能扩大成  $[-1, 0]$ ,对  $p_e = 0$  的差值进行扩展嵌入才不会引起内外区域的重叠。需说明的是,本轮循环只是搜索,并不会真的进行嵌入操作。按照溢出条件(随后讨论)对将要扩展嵌入的差值和平移的差值进行判断,把可能引起图像像素值溢出的位置进行标示,创建溢出位置图。在除开溢出点个数后,计算各个可扩展差值的  $f(p_e)$ ,得到总的可嵌入点个数  $count$ ,然后判断是否满足容量条件:

$$count \geq \eta(P) + \eta(Q) + \eta(M) \quad (1)$$

式中,  $\eta(P)$ ,  $\eta(Q)$  和  $\eta(M)$  分别是负载、头文件和压缩溢出图的大小。如果式(1)满足,则表明已经找到足够的待嵌入差值位置,结束这个寻找循环。如果不满足,开始完全新一轮的循环,此时除了  $p_e = 0$  的差值,还要考虑以  $p_e = 1$  的差值进行扩展。在新一轮的循环中,左右阈值分别变为  $T_n = 1, T_p = 1$ ,除了重复上述直方图左侧的操作外,还要在直方图正半轴(内区域以外)选择一个最靠近原点的零点  $Z_{p1}$ ,对外区域  $[T_p + 1, Z_{p1} - 1]$  右移一个单位,变成  $[T_p + 2, Z_{p1}]$ ,这样内区域才能扩大成  $[-1, 2]$ ,对  $p_e = 0, 1$  的差值进行扩展嵌入才不会引起内外区域的重叠。同理,依照在直方图左边处理的方法,标示可能产生溢出的差值位置,继而计算  $p_e = 0, 1$  位置上的可嵌入点个数  $count$ ,判断是否满足式(1)。如果不满足,则再开始完全新一轮的循环,此时,除了  $p_e = 0, 1$  的差值,还要考虑以  $p_e = -1$  的差值进行扩展。此时,  $T_n = 2, T_p = 1$ 。在直方图负半轴(内区域以外)选择最靠近和次靠近原点的零点  $-Z_{n1}$  和  $-Z_{n2}$ ,则负半轴外区域分别为  $[-Z_{n1} + 1, -T_n]$  和  $[-Z_{n2} + 1, -Z_{n1} - 1]$ ,经过平移后,分别变成

$[-Z_{n_1} - 1, -T_n - 2]$ 和 $[-Z_{n_2}, -Z_{n_1} - 2]$ 。右半轴还是与新一轮循环相同,做相同的处理。这样内区域才能扩大成 $[-3, 2]$ 。同理,依照上述方法,标示可能产生溢出的差值位置,继而计算 *count*, 判断是否满足容量条件。如此在直方图两侧交替选择,直至最终满足容量条件。可以看到,最终所得到的内区域为 $[-T_n + 1, T_p]$ 。

2.1.3 零点的使用以及直方图的分段和平移

本节详细介绍如何用零点对直方图的外部区域进行分段。用一般形式表示,具体操作由  $T_n$  决定。如图 3 所示,这里的零点距离横轴原点由近及远分

别为: $-Z_{n_1}, -Z_{n_2}, \dots, -Z_{n_{T_n}}$ 。把外部区域 $[-Z_{n_{T_n}} + 1, -T_n]$ 分成了  $T_n$  个区间。各区间平移的规则是:第一区间 $[-Z_{n_1} + 1, -T_n]$ 左移的距离为  $T_n$ ;第二区间 $[-Z_{n_2} + 1, -Z_{n_1} - 1]$ 左移的距离为  $T_n - 1$ ;...;第  $T_n$  区间 $[-Z_{n_{T_n}} + 1, -Z_{n_{T_n-1}} - 1]$ 左移的距离为 1。这样,就把 $[-2T_n + 1, -T_n]$ 的区间清空了。可以看到,这比把整个左边外部区域当成一个整体平移  $T_n$  对图像质量的影响要小很多,而 Thodi 等人的算法正是进行整体位移。同理,在正半轴的直方图分段与平移和负半轴的类似,最终把 $[T_p + 1, 2T_p]$ 的区间清空,为扩展嵌入做准备。

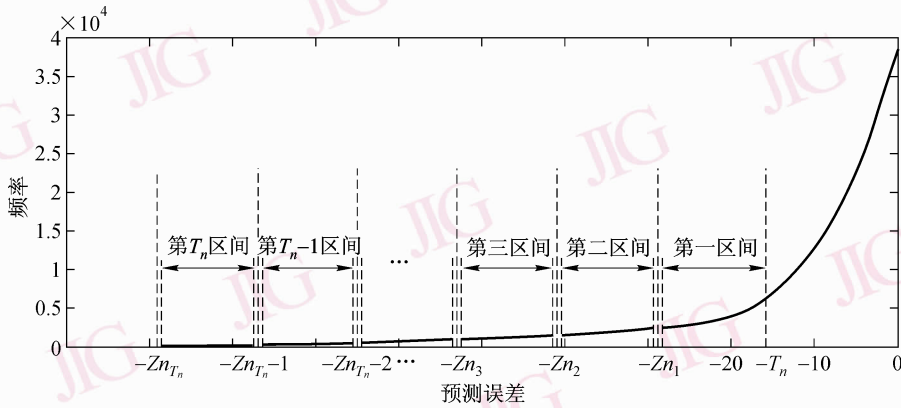


图 3 负半轴直方图分段  
Fig. 3 Histogram segmentation on negative axis

2.1.4 嵌入公式

若首次嵌入扩展方向为负向,则对于内区间的差值  $P_e \in [-T_n + 1, T_p]$  来说,嵌入公式为

$$p'_e = 2p_e - i \quad (2)$$

若首次嵌入扩展方向为正向,则嵌入公式为

$$p'_e = 2p_e + i \quad (3)$$

嵌入公式的选取主要是为了适应首次扩展方向,使用式(2)或式(3)本身对算法并没有根本影响。

2.1.5 提取公式

嵌入后  $p_e$  的范围变成  $p'_e \in [-2T_n + 1, 2T_p]$ ,那么相应的数据提取和原像素恢复公式有如下两种情况:

当  $-2T_n + 1 \leq p'_e \leq 0$  且溢出位置图上该差值对应的位置标记为“0”,则

$$\begin{aligned} i &= |p'_e| \% 2 \\ p_e &= -\lfloor |p'_e| / 2 \rfloor \\ x &= \hat{x} + p_e \end{aligned} \quad (4)$$

当  $1 \leq p'_e \leq 2T_p$  且溢出位置图上该差值对应的位置标记为“0”,则

$$\begin{aligned} i &= |p'_e| \% 2 \\ p_e &= \lfloor |p'_e| + 1 / 2 \rfloor \\ x &= \hat{x} + p_e \end{aligned} \quad (5)$$

2.1.6 溢出条件和溢出位置图的建立

如前所述,对全体差值的操作有两类:嵌入和平移,两种操作都可能导致原 8 比特图像的像素值超过  $0 \sim 255$  的范围,即  $x' \notin [0, 255]$ ,因此其边界条件分为两类:

对于嵌入,可能发生溢出的位置有:

$$\begin{cases} x < 1 & p_e = 0 \\ 2x - \hat{x} < 1 & -T_n + 1 \leq p_e \leq -1 \\ 2x - \hat{x} > 255 & 1 \leq p_e \leq T_p \end{cases} \quad (6)$$

对于平移,则需要保证外区域的像素值加上该段的平移位移后大于等于 0 且小于等于 255。

凡是可能导致溢出的差值位置,将在溢出图中

用“1”标识出来,并不进行嵌入或平移操作,而其他位置则用“0”标识。每次使用溢出位置图前,先用二进制“0”进行初始化。值得指出的是,本文的溢出位置图只记录在确定的差值选择阈值  $T_n$  和  $T_p$  下内外区域差值可能溢出的状态。很显然,本文的溢出位置图是依赖于负载的,因为负载确定  $T_n$  和  $T_p$  的大小。在进行数据抽取和图像恢复的时候,溢出位置图必须和阈值  $T_n$  和  $T_p$  联合使用才能判断当前差值的状态。本文的溢出位置图与 Thodi 等人算法的溢出位置图有本质的不同。

2.1.7 头文件

头文件  $Q$  是为了可逆地恢复原图像数据而设置的,记录了全部的嵌入信息。 $Q$  由下面的几部分组成:直方图首次扩展的方向标志位  $F$  (占 1 比特),  $T_n$  值 (占 8 比特),直方图负半轴上的零点值 (占  $8T_n$  比特),  $T_p$  值 (占 8 比特),直方图正半轴上的零点值 (占  $8T_p$  比特),最后一个嵌入点的坐标  $(x, y)$  (占 32 比特),压缩溢出图的长度  $\eta(M)$  (占 24 比特)。

2.1.8 预测器

本文算法采用了图像预测编码中两个常用的预测器,一个是反因果预测器,另一个是因果加权平均预测器,图 4 表示各像素点间的位置关系。

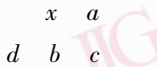


图 4 各像素点间的位置关系  
Fig. 4 Context of a pixel

反因果预测器利用 3 个上下文像素值进行预测,预测算法为<sup>[6]</sup>:

$$\hat{x} = \begin{cases} \max(a, b) & c \leq \min(a, b) \\ \min(a, b) & c \geq \max(a, b) \\ a + b - c & \text{其他} \end{cases} \quad (7)$$

因果加权平均预测器利用 4 个上下文像素值进行预测,预测算法为<sup>[8]</sup>:

$$\hat{x} = (2a + 2b + c + d) / 6 \quad (8)$$

预测器的选取关系到差值图像的获取,好的差值预测器会去掉原始图像像素间大部分相关的信息,从而使得差值分布集中在 0 附近,这样尖锐的直方图无疑会提高算法的性能,但预测器的设计超出了本文的研究范围,这里只以常用的预测器来展示本文算法的性能。实际上,本文算法适合于任何差值图像。

2.2 嵌入过程

可利用上述预测器以光栅扫描序扫描原始图像获得预测误差图像。有了差值图像即可获得预测误差的直方图。对于给定的负载,利用 2.1.1 节介绍的方式即可确定直方图外部区间首次扩展的方向,并确定与之匹配的嵌入公式。数据隐藏的嵌入过程分为 2 个阶段:第 1 个阶段选择用于嵌入的区域,第 2 个阶段进行真正的数据隐藏嵌入。

2.2.1 选择嵌入区域的流程图

用图 5 来解释如何平移直方图创造空闲空间以及如何确定待嵌入的差值位置。

图 5 所描述的过程设首次差值扩展方向为左,在首次差值扩展方向为右情况下可以得出类似过

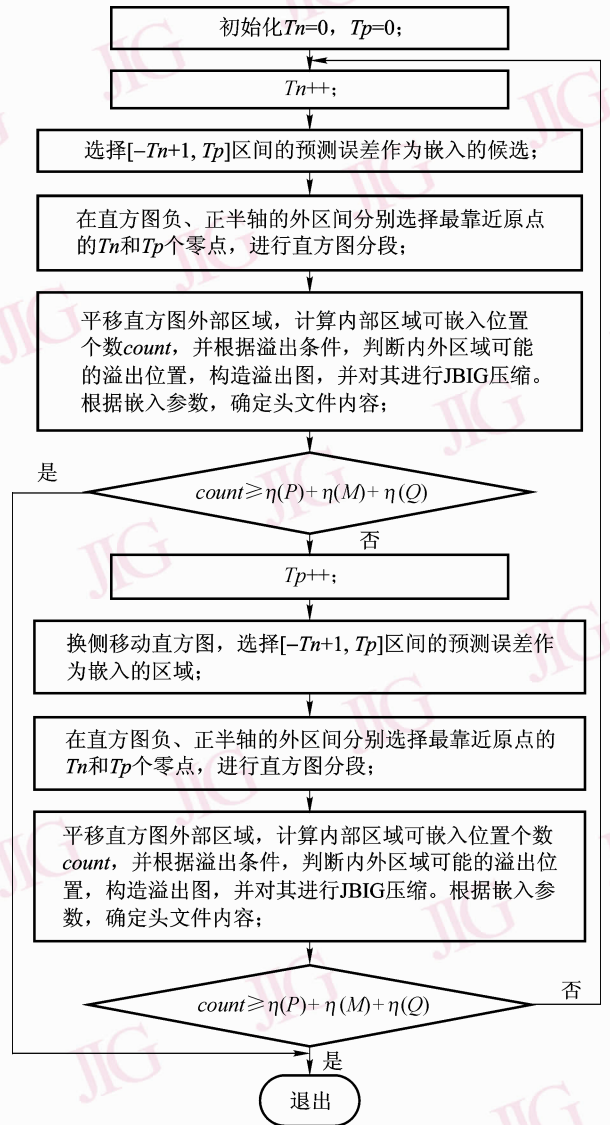


图 5 选择嵌入区域的流程图

Fig. 5 Flow chart of choosing embedded region

程,不再赘述。

### 2.2.2 数据隐藏嵌入

当第1个阶段结束时,得到了头文件 $Q$ 和压缩后的溢出位置图 $M$ ,因此,实际要隐藏嵌入的数据比特流由三部分组成: $Q+M+P$ 。接下来,传统的做法是依次将这三部分嵌入到第1阶段所确定嵌入位置中。然而,在数据盲抽取时,这种嵌入方式会导致一个问题:如果不知道头文件和溢出位置图,如何进行盲数据抽取?因此,必须设计一种在盲数据抽取时可以先得到头文件和溢出位置图信息的算法。首先要把存储的三个部分调整次序,变成: $P+Q+M$ ,然后借鉴文献[6]的做法,将 $Q+M$ 的信息直接保存在最后输出图像的前 $\eta(Q)+\eta(M)$ 个像素的LSB(least significant bit)位,其顺序仍从左上角按光栅扫描序算起。而原来这些位置上的LSB位则被存放在准备用于存储 $Q+M$ 数据流的位置。换言之,这两个部分只是交换了一下存储的地方,没有损失任何信息。具体的操作细节可以参见文献[6]。值得指出的是,本文算法具有精确控制容量的能力,因此,把最后一个嵌入位置的坐标记录下来,保存在头文件中。

### 2.3 提取过程

由于预测器采用的是后向预测,所以以逆光栅扫描序扫描图像,即数据的抽取过程与嵌入过程步骤刚好相反。数据抽取的具体步骤如下:

(1)从检测图像的前 $\eta(Q)+\eta(M)$ 像素的LSB比特提取出头文件信息 $Q$ 和压缩的溢出图,具体顺序为:提取第一个比特为标志位 $F$ ,接下来的8比特构成 $T_n$ 值,再接下来的 $8T_n$ 比特构成直方图负半轴上的零点值,再后的8比特构成 $T_p$ 值,再后的 $8T_p$ 比特构成直方图正半轴上的零点值,接着的32比特构成最后一个嵌入点的坐标 $(x,y)$ ,再接着的24比特构成压缩溢出图的长度 $\eta(M)$ ;得到 $\eta(M)$ 后,我们紧接着提取 $\eta(M)$ 个LSB,构成压缩后的溢出图;

(2)对压缩的位置图进行JBIG解压缩,得到溢出位置图 $M$ ;

(3)开始恢复图像:逆光栅扫描图像,从头文件记录的最后一个嵌入点坐标开始,计算预测值 $\hat{x}$ ,求出预测误差 $p'_i = x' - \hat{x}$ ,若预测误差嵌入了比特,则提取嵌入的比特 $i$ ,并恢复原预测误差,马上用 $i$ 来替换图像第 $\eta(P)+\eta(M)$ 个位置上像素的LSB,一直到恢复完前 $\eta(P)+\eta(M)$ 个位置上像素的LSB比特为止。若预测误差进行了平移,则进行逆操作,恢复原来的预测误差;否则,保持预测误差不改变。

(4)重复步骤3的操作,直到恢复所有的预测误差;

(5)计算原像素 $x' = \hat{x} + p'_e$ ,得到原图像。

## 3 实验结果

为了验证本文算法的性能,对一些标准测试图像进行了实验,包括Lena, F16, Baboon, Girl, Stream and bridge,如图6所示。在图7中给出4幅典型的直方图。为了比较,还给出了Thodi等人的P2算法[6]的实验结果以及Tian算法的结果,如图8所示。Thodi等人在文献[6]中给出了算法的几个版本,其中P2算法的总体性能最好。



图6 部分标准测试图

Fig. 6 Some test images

从图8(a)、(b)看出,本文算法比Thodi等人的方法略好,但图像质量提高的不太大,这主要是因为Lena和F16的直方图比较尖锐。对于普通具有尖锐直方图的图像,本文算法的优势在于利用了零点和依赖于负载的溢出位置图,所以,在中、低嵌入率(bpp)下,图像的峰值信噪比(PSNR)(单位:dB)有所提高(如图8(b))。

本文算法的主要优势体现在改善纹理图像和直方图有拖尾的图像。从图8(c)、(d)看到,本文方法的结果大大好于Thodi等人的结果,这主要是因为Baboon图像的直方图比较平坦,而Girl的直方图有拖尾。此时,本文方法的三个优点都可以反映出来:单侧移动直方图可以大大减少不必要的差值平移;零点可以将有拖尾直方图分成很多段,对于某一负载,只需要移动相应的几段,从而减少平移的位移;而依赖于负载的溢出位置图极大地提高了压缩的效率,从而减小了压缩后溢出位置图的尺寸。

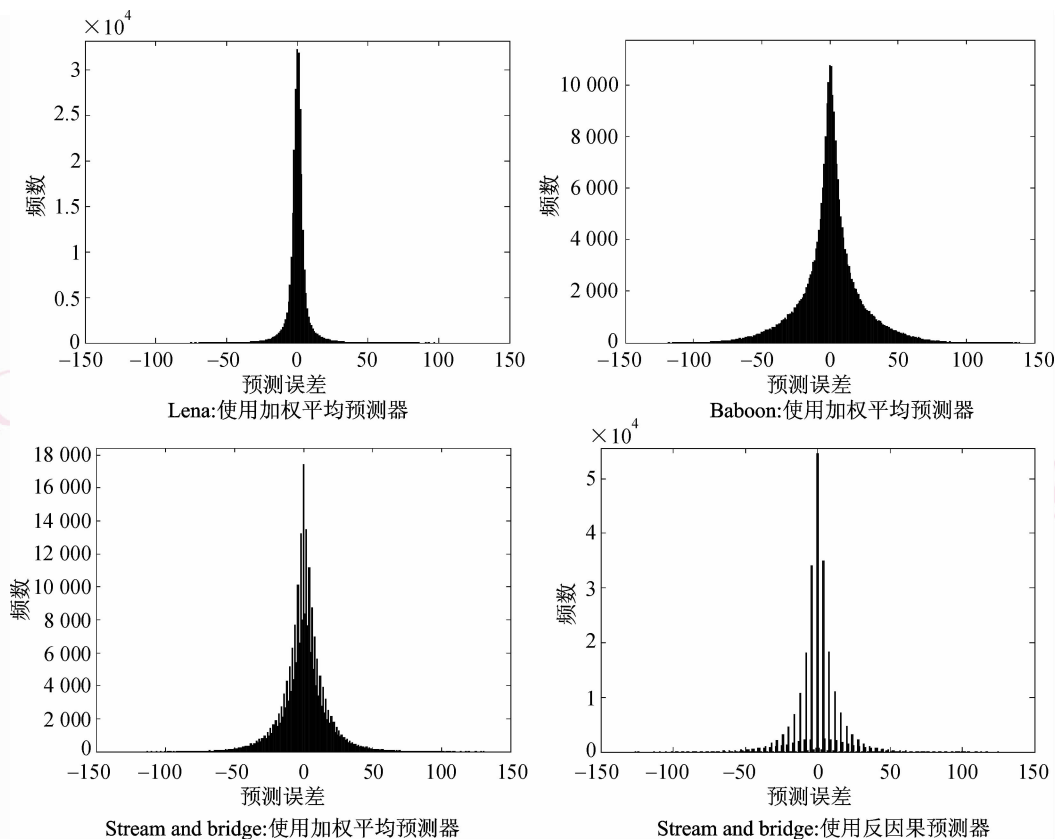


图 7 测试图像的直方图  
Fig. 7 Histogram of test images

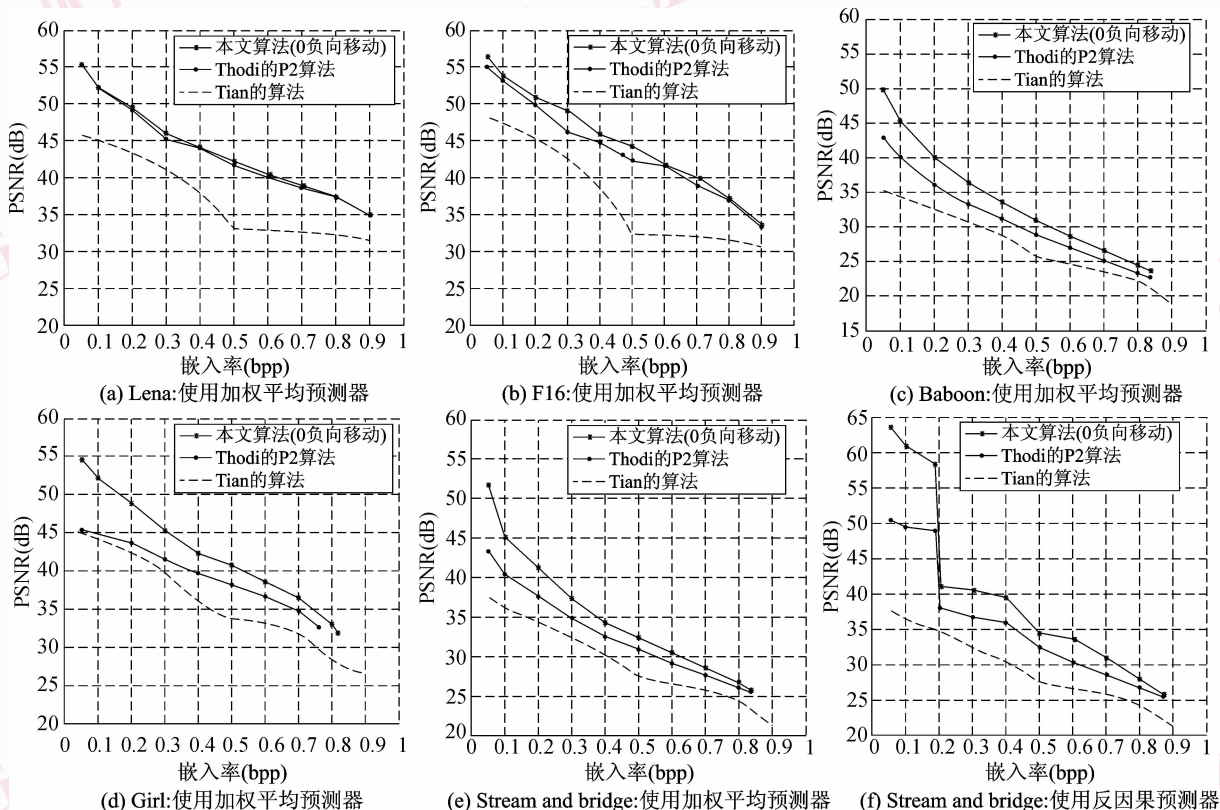


图 8 本文算法和 Thodi 的 P2 算法, Tian 的算法的试验结果

Fig. 8 Experimental results of our method, Thodi's P2 algorithm and Tian's algorithm

为了说明零点的作用,给出一个极端的例子,对 Stream and bridge 图像而言,从其使用反因果预测器的直方图可以看到,有非常多的零点存在,而且有些零点距离原点不远,所以对算法性能有明显影响。在嵌入率小于 0.2 的时候,输出图像质量比 Thodi 等人的高出很多,这主要是使用零点后直方图分段平移的位移减小。随着嵌入率的增加,因为内区域的零点不能使用,所以直方图平移的位移有所增加,故算法性能有所下降。当嵌入率很高时,本文算法的性能逐渐和 Thodi 等人的算法性能相近,这是由于此时的零点大都在直方图的尾端,直方图平移造成的变形和 Thodi 的差不多。

可以看到,在以上所有的情况下,本文算法的结果都大大优于 Tian 的经典可逆嵌入算法。

## 4 结 论

本文提出了一种在图像差值中进行差值扩展嵌入的可逆数据隐藏技术,针对现有文献的不足,主要提出了 3 种策略改进算法的性能。实验证明,提出的交错直方图平移、零点分段直方图以及依赖于负载的溢出位置图很好地利用了不同差值图像直方图的特点,在不降低嵌入率的条件下有效地改善了输出图像的质量。本文算法的结果比文献中具有代表性的最新算法的结果要好,并且,在中、低嵌入率下,效果更为明显。值得说明的是,尽管本文算法是基

于预测误差直方图所提出的,但其思想适合于其他的差值直方图。

## 参考文献 (References)

- 1 Fridrich J, Goljan M, Du R. Lossless data embedding-new paradigm in digital watermarking [J]. EURASIP Journal on Applied Signal Processing, Special Issue on Emerging Applications of Multimedia Data Hiding, 2002, **2002**(2):185-196.
- 2 Tian J. Reversible data embedding using a difference expansion[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2003, **13**(8): 890-896.
- 3 Celik M U, Sharma G, Tekalp A M, *et al.* Lossless generalized-LSB data embedding[J]. IEEE Transactions on Image Processing, 2005, **12**(2):157-160.
- 4 Alattar A M. Reversible watermark using the difference expansion of a generalized integer transform [J]. IEEE Transactions on Image Processing, 2004, **13**(8):1147-1156.
- 5 Kamstra L, Heijmans H J A M. Reversible data embedding into images using wavelet techniques and sorting[J]. IEEE Transactions on Image Processing, 2005, **14**(12):2082-2090.
- 6 Thodi D M, Rodriguez J J. Prediction-error based reversible watermarking[J]. In: Proceedings of IEEE International Conference on Image Processing[C], Singapore, 2004, **3**: 1549-1552.
- 7 Thodi D M, Rodriguez J J. Expansion embedding techniques for reversible watermarking[J]. IEEE Transactions on Image Processing, 2007, **16**(3):721-730.
- 8 Yip Shu-kei, Au O C, Wong Hoi-Ming, *et al.* Generalized lossless data hiding by multiple predictors [A]. In: Proceedings of IEEE International Symposium on Circuits and Systems[C], Island of Kos, Greece, 2006:1426-1429.