

# 基于直方图对二值图像无损数据隐藏算法的改进

童学锋<sup>1</sup>, 侯海锋<sup>1</sup>, 宣国荣<sup>1</sup>, 施云庆<sup>2</sup>

(1. 同济大学 电子与信息工程学院, 上海 201804; 2. 新泽西州工学院 电气工程和计算机系, 美国 新泽西 07102)

**摘要:** 提出一种改进基于游程编码和直方图对二值无损数据隐藏算法的新方法. 先对载体二值图像进行“与”操作或“异或”操作的预处理, 再利用基于游程编码的直方图对的二值图像无损数据隐藏方法进行数据嵌入. 实验表明, 该方法在嵌入容量和视觉效果都优于改进前的方法.

**关键词:** 预处理; 二值图像; 无损数据隐藏; 游程编码; 直方图对

中图分类号: TP 391.41

文献标识码: A

## Improvement of Reversible Binary Image Data Hiding by Run-length Coding and Histogram Pair

TONG Xuefeng<sup>1</sup>, HOU Hai Feng<sup>1</sup>, XUAN Guorong<sup>1</sup>, SHI Yunqing<sup>2</sup>

(1. College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China; 2. Department of Electrical and Computer Engineering, New Jersey Institute of Technology, New Jersey 07102, USA)

**Abstract:** The paper presents a novel method which enhances the capability of the reversible binary image data hiding scheme using run-length (RL) histogram modification algorithm. Before adopting the reversible binary image data hiding scheme using histogram RL modification algorithm, some pretreatments like AND operation and XOR operation are processed first. Experimental results prove its superiority to other methods.

**Key words:** pretreatment; binary image; reversible data hiding; run-length; histogram pair

图像无损数据隐藏是近年来一个新的研究方向, 在这个领域发表的论文数量逐年增多, 但对于二

值图像无损数据隐藏, 有关文章不多<sup>[1-4]</sup>. 文献[4]提出的基于游程编码的直方图对的二值图像无损数据隐藏方法在解决文字和图表两者、或半色调和非半色调两者的兼顾问题上, 有一定的优越性. 本文在基于游程编码的直方图对的二值图像无损数据隐藏方法<sup>[4]</sup>的基础上进行了一些改进.

为了提高二值图像数据隐藏的性能, 采用逻辑运算预处理, 预先去除相关, 使黑游程的分布变尖, 黑游程的分布的熵减少, 冗余增加. 这样一来, 嵌入数据时, 像素改变减少, 嵌入最大容量增加. 实验表明, 改进后的二值图像数据隐藏方法在嵌入容量和视觉效果方面比原算法好.

## 1 基于游程编码和直方图对方法的数据嵌入

游程编码, 又称行程长度编码, 是一种无损压缩编码. 例如对于一段二值数据“1100011100111100”, 用游程编码可以表示为: 233242, 表示为游程对则为:  $\langle 2, 3 \rangle, \langle 3, 2 \rangle, \langle 4, 2 \rangle$ . 这种编码可以把二值数据用多值来表示, 由此便可以用直方图对<sup>[5]</sup>的思想来实现无损数据隐藏.

### 1.1 基于游程编码的数据嵌入的简单例子

#### 1.1.1 数据嵌入

图像的扫描顺序是从左到右, 从上到下, 由此形成黑游程和白游程长度交替序列. 也就是说, 第一行尾巴连接到第二行的头形成一维序列, 由一个黑游程, 紧跟一个白游程, 组成游程对, 最后形成游程对序列. 图1给出了2行13列的二值图像, 其中右半部分的0和1分别表示白像素和黑像素, 而左半部分则给出了黑白游程长度、图像的游程对表示; 按上

收稿日期: 2010-01-15

基金项目: 国家自然科学基金(90304017)

第一作者: 童学锋(1963—), 男, 教授, 博士生导师, 工学博士, 主要研究方向为图像处理、模式识别. E-mail: tongxuefeng@tongji.edu.cn

下顺序可以分为 3 部分,其中(I)为原图,(II)为嵌入过程中创建直方图对后的图,(III)为嵌入数据后的图.

从图 1 中的(I)可知,黑游程的直方图为: $h(1) = 3, h(2) = 1, h(3) = 2$ . 假如嵌入位置  $T = 1$ ,即选取长度为 1 的黑游程为嵌入位置,因为  $h(1) = 3$ ,因此可以嵌入 3 bit 数据. 为了嵌入数据首先需要创建直方图对<sup>[6]</sup>, $[h(1) = 3, h(2) = 0]$ ,为此,需扫描图像,找出值为 2 的黑游程,即两个连续的黑像素点. 在第 1 行找到一个长度为 2 的黑游程(第 8, 9 列),然后扩展黑游程长度使之从 2 到 3,这样就使得  $h(2) = 0$  和  $h(3) = 3$ . 这个扩展需要记录(bookkeeping)下来以便将来能无损恢复原始图像. 注意到这种操作使得跟在黑游程后面的白游程的值减 1,因此游程对中黑游程的值与白游程的值之和并未改变. 图 1 中的(II)给出了修改后的图像、游程对以及黑白游程直方图对.

游程→	1	2	3	列 →	1	2	3	4	5	6	7	8	9	10	11	12	13	
(I) 原始二值图像: <1,2>,<1,3>,<2,3> <1,2>,<3,3>,<3,2>	黑游程 直方图	3	1	2	第 1 行	1	0	0	1	0	0	0	1	1	0	0	0	1
	白游程 直方图	0	3	3	第 2 行	0	0	1	1	1	0	0	0	1	1	1	0	0
(II) 创建直方图对 [3,0] <1,2>,<1,3>,<3,2> <1,2>,<3,3>,<3,2>	黑游程 直方图	3	0	3	第 1 行	1	0	0	1	0	0	0	1	1	0	0	0	1
	白游程 直方图	0	4	2	第 2 行	0	0	1	1	1	0	0	0	1	1	1	0	0
(III) 嵌入数据:1,0,1; $T=1$ <2,1>,<1,3>,<3,2> <2,1>,<3,3>,<3,2>	黑游程 直方图	1	2	2	第 1 行	1	0	1	0	0	0	1	1	0	0	0	0	1
	白游程 直方图	2	2	2	第 2 行	0	1	1	1	0	0	0	1	1	1	1	0	0

图 1 一个简单例子

Fig. 1 A simple example

直方图对创建好以后,就准备嵌入数据. 由于  $h(1) = 3$ ,可以嵌入 3 bit 数据,假设需要嵌入的数据为 1,0,1,顺序扫描(II)右边表示的二值图像;首先扫描到的黑游程的长度为 1(第 1 行,第 1 列),检查第一个需要嵌入的比特为 1,改变第 1 行第 2 列的值从 0 到 1,则黑游程长度从 1 变为 2,对应的白游程长度缩短 1;接着扫描到下一个单个黑像素(第 1 行,第 4 列),而此时需嵌入的比特为 0,所以不改变;继续扫描遇到的第 3 个单个黑像素(第 1 行,第 13 列),而需嵌入比特为 1,改变第 2 行第 1 列的值从 0 到 1. 二值图像嵌入的最后结果表示在(III)的右边,游程对和它的直方图表示在(III)的左边.

### 1.1.2 数据提取

数据提取从嵌入数据后的图像开始,它表示在(III)的右边. 用与嵌入相同的顺序来检查像素的变化. 如果遇到黑游程长度为 1,抽取数值 0;如果遇到

黑游程长度为 2,抽取数值 1,同时将黑游程长度还原为 1. 注意记录(bookkeeping)数据用来无损恢复图像.

## 2 预处理

1.1 节描述的方法已经可以取得较好的嵌入效果,但从图像的性质可知,图像相邻两行存在很大的相关性,如果在嵌入操作之前首先去除相邻两行的相关性,将使得嵌入容量更大而且视觉效果更好. 因此通过预处理,在嵌入运算前预先去除相关性,这样一来,在嵌入数据时,像素改变变少,嵌入最大容量也增加.

预处理的基本操作是把图像分为奇数行和偶数行两部分,嵌入操作只在偶数行部分进行,奇数行不变;把每一个奇数行与对应的下一偶数行进行逻辑运算得到新的偶数行,把所有新的偶数行按先后顺序连成一个比特流串,然后在这样的比特串上应用前面介绍的方法进行数据隐藏操作. 本文则使用了以下两种逻辑操作:

(1)“与”运算.“与”运算解相关,实际上是指定窗口[100000...]的奇偶数行“与”运算. 假设奇数行不变,偶数行嵌入. 如果在奇数行观察到这样的模式,那么可以认为偶数行在相应的位置出现这样的模式的概率最大. 也就是说,偶数行相应位置适宜用于嵌入的概率最大. 这时将偶数行相应位置的图像选中.“与”运算有些类似于多值图像中预测方法的预测作用,使冗余减少,分布变尖,熵降低,易于数据的嵌入.

(2)“异或”运算. 实质上两行之间各元素“异或”运算是一种模变换,如果设 $\oplus$ 表示“异或”, $(\cdot)_2$ 表示模 2 运算,则有  $x_1 \oplus x_2 = (x_1 - x_2)_2 = (x_1 + x_2)_2$ . “异或”运算与小波变换很相似,小波正变换: $l = x_1, h = (x_1 - x_2)_2$ ,小波逆变换: $x_1 = l, x_2 = (h + l)_2$ . 小波高频子带有解相关作用,使冗余减少,分布变尖,熵降低,易于数据的嵌入.

### 2.1 基于“与”运算的二值无损数据隐藏

为了提高数据隐藏的容量,提出在数据嵌入前对载体图像进行“与”操作处理. 在数据抽取和原图恢复中需要增加逆操作.

图 2 中(I)为原图,已经将图中的孤立白点去除<sup>[4]</sup>. 以下为运用于该图的嵌入过程.

(1) 将载体图像划分成奇数行和偶数行两部分,分别用  $X_1$  和  $X_2$  表示. 在这个例子中  $X_1 = x_1$ ,

$X_2 = x_2$ . 在数据嵌入的过程中, 奇数行部分固定不变, 偶数行嵌入数据.

(2) 扫描  $x_1$  行, 选出所有符合一个黑像素点后面紧跟若干个白像素点这一模式的所有像素组.

(3) 对(2)中找出的所有像素组, 在紧跟的下一行(本例中为  $X_2$ )中选出与之对应的像素组. 可以构建一个一维窗口, 窗口的位置、长度和(2)中选出的像素组完全一样, 然后将这个窗口和下一行( $X_2$ )做“与”操作.

(4) 对  $x_2$  行中每个被选出的像素组, 去除刚开始连续的白像素点和末位连续的黑像素点.

(5) 将(4)中处理后的像素组按从上到下, 从左到右的顺序连接成一个一维像素串.

(6) 通过游程组技术形成(4)中序列所有游程组的直方图. 注意: 这个直方图一般是多值的.

(7) 利用直方图对方法嵌入数据.



图 2 基于“与”运算的二值无损数据隐藏实例

Fig.2 Example of RL histogram modification involving AND operation

(8) 数据嵌入后, 将嵌入数据的偶数行部分和未改变的奇数行合并形成嵌入后的图像.

(9) 数据提取的过程与数据嵌入的过程刚好相反.

载体二值图像, 每个步骤后的图像以及相应的游程组直方图对全部显示在图 2 中.

在图 2 中, (I) 将原图分成奇数行部分和偶数行部分. 在奇数行中设置一个与模式  $[1, 0, 0, \dots, 0]$  对应的窗口  $w = [1, 1, \dots, 1]$ ; (II) 选出所有类似  $[1, 0, 0, \dots, 0]$  模式的像素组, 进行“与”操作:  $x'_2 = x_2 \& w$  ( $x'_2$  的外面是  $x_3$ ); (III) 创建  $x'_2$  的游程对序

列. (IV) 在阈值  $T$  处嵌入数据, 得到  $x''_2$ ; (V) 与固定的奇数行组合成嵌入后的图像.

## 2.2 基于“异或”运算的二值无损数据隐藏

图 3 中(I)为原图, 已经将图中的孤立白点去除; (II)为“异或”操作; (III)为游程对编码; (IV)为创建直方图对; (V)为在阈值  $T$  为处嵌入数据; (VI)为组合成嵌入后的图像.



图 3 基于“异或”运算的二值无损数据隐藏

Fig.3 Example of RL histogram modification involving XOR operation

二值图像的嵌入方法的流程如下所示:

(1) 将载体图像划分成奇数行和偶数行两部分, 分别用  $X_1$  和  $X_2$  表示. 在这个例子中  $X_1 = x_1$ ,  $X_2 = x_2$ .

(2) 对于  $X_2$  中的每一行, 对这一行和它的紧靠的上一行( $X_1$ )进行“异或”运算.

(3) 将(2)中处理后的数据保持从上到下的顺序, 形成一个二维的数组  $Y$ .

(4) 忽略开头的连续白像素点和末尾的连续黑像素点, 按照从上到下, 从左到右的顺序扫描  $Y$ .

(5) 应用游程直方图对无损数据嵌入的方法, 得到另一个二维数组  $Z$ .

(6) 将  $X_2$  中的奇数行替换到  $Z$  中的相应的行中.

(7) 将  $X_1$  和  $Z$  组合起来(将  $X_1$  作为偶数行部分,  $Z$  作为相应的奇数行部分)形成嵌入后的图像.

二值图像的提取流程如下:

(1) 将给定的二值图像  $X$  分成奇, 偶两部分, 分别用  $X_1$  和  $X_2$  表示.

(2) 对于  $X_2$  中的每一行,在这一行和它的紧跟上一行( $X_1$ )之间运用“异或”操作.

(3) 将(2)中得到的结果按从上到下的顺组成一个二维数组  $Y$ .

(4) 忽略开头的连续白像素点和末尾的连续黑像素点,按照从上到下,从左到右的顺序扫描  $Y$ .

(5) 运用游程对直方图抽取数据的方法得到一个二维数组  $Z$ .

(6) 将  $X_2$  中的奇数行数据替换到  $Z$  中的相应行中.

(7) 将  $X_1$  和  $Z$  组合起来(将  $X_1$  作为偶数行部分,  $Z$  作为相应的奇数行部分)形成嵌入后的图像.

### 3 实验结果

#### 3.1 简单非半色调图像

图 4a 所示的一幅大小为 274 像素  $\times$  312 像素的米老鼠二值图像中,嵌入 128 bit,引起 137 个像素改变,其中记录量为零.原图、嵌入后图像和表示比特改变的图像,分别见图 4a—4c.

米老鼠二值图像的更多实验结果见表 1.

表 1 274 像素  $\times$  312 像素米老鼠非半色调二值图像结果  
Tab.1 Mickey non-halftone binary image of 274  $\times$  312

嵌入容量/bit (不包括记录)	改变的像素个数		
	未预处理	“与”处理	“异或”处理
100	96	72	98
128	137	105	159
200	245	197	287
432	653	524	834

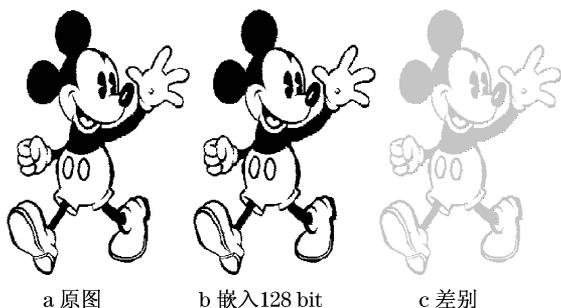


图 4 274 像素  $\times$  312 像素米老鼠二值图像  
Fig.4 Mickey binary image of 274  $\times$  312

#### 3.2 普通复杂二值图像

图 5 为一幅 baboon.bmp 512 像素  $\times$  512 像素(通过 MATLAB 的 im2bw 指令得到的阈值化)普通二值图像.对其嵌入数据 1 063 bit,共有 1 901 个像素发生变化,其中记录量为 230 bit.

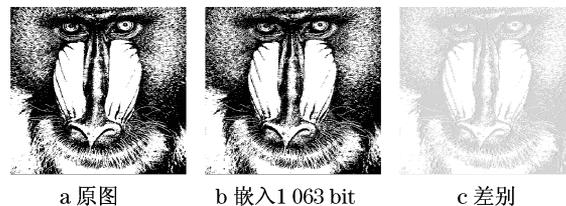


图 5 512 像素  $\times$  512 像素普通狒狒二值图像  
Fig.5 Baboon non-halftone binary image of 512  $\times$  512

半色调狒狒二值图像的更多实验结果见表 2.

表 2 512 像素  $\times$  512 像素狒狒普通二值图像结果

Tab.2 Baboon non-halftone binary image of 512  $\times$  512

嵌入容量/bit (不包括记录)	改变的像素个数		
	未预处理	“与”处理	“异或”处理
100	78	57	45
200	169	136	116
300	286	232	179
409	398	349	284
516	554	510	433
641	794	714	635
783	1 109	949	807
867	1 387	1 116	948
1 063	1 901	1 637	1 259

#### 3.3 半色调二值图像

用 MATLAB 的 dither 形成的大小为 512 像素  $\times$  512 像素的半色调狒狒二值图像,如图 6a 所示,在嵌入 7 699 bit 时,会引起 6 787 个像素的改变,其中记录量为 1 491 bit.实验的原图、嵌入后图像和表示比特改变的图像,分别见图 6a—6c.

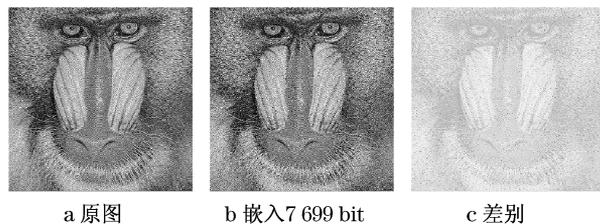


图 6 512 像素  $\times$  512 像素狒狒半色调二值图像  
Fig.6 Baboon halftone binary image of 512  $\times$  512

狒狒半色调二值图像的更多实验结果见表 3.

#### 3.4 预处理前后效果比较

从表 1 中可以看出,经过“与”运算预处理后,对于相同的嵌入容量,改变的像素个数明显要少.当不进行任何预处理,嵌入容量为 128 bit 时,图像改变的像素个数为 137 个,经过“与”运算预处理后,改变的像素个数仅为 105,比前者有较大的改进.但是“异或”预处理的效果不是很理想.

而对于半色调二值图像,如表 2 所示,“与”运算

预处理和“异或”运算预处理都获得了很好的效果,并且“异或”运算效果要好于“与”运算.同样嵌入800 bit的数据,未处理前改变的像素个数为537,通过“与”运算处理后改变的个数则减少到484,而经过“异或”运算后,效果则更佳,只改变了408个像素.

从以上的实验数据可以看出,“与”运算的预处理方法对纹理简单的二值图像,如米老鼠,比较有效;“异或”运算的预处理方法对纹理复杂的二值图像,如狒狒二值图像,比较有效.

表3 512像素×512像素狒狒半色调二值图像结果  
Tab.3 Baboon halftone binary image of 512×512

嵌入容量/bit (不包括记录)	改变的像素个数		
	未预处理	“与”处理	“异或”处理
400	241	212	203
800	537	484	408
1 600	1 123	998	866
2 400	1 706	1 589	1 327
3 200	2 356	2 117	1 873
4 000	3 111	2 796	2 341
4 800	3 964	3 347	2 877
5 600	4 483	4 012	3 749
6 400	5 990	5 347	4 871
7 699	6 787	6 345	5 692

### 3.5 预处理的结果分析

“与”运算:处理方法是奇数行的符合“100...”模式的二进制串选出来,由于图像的连续性和相关性,则在偶数行对应位置的二进制串最有可能出现符合“100...”模式的情况,那么对于黑白游程对编码方式, $\langle 1, n \rangle$ 这种情况的游程组就会比较多,直方图统计下来黑游程为1处的直方图会比较高,整个直方图统计在横坐标为1处就比较尖,比较有利于利用直方图对方法进行嵌入.如图7所示,米老鼠二值图像在进行“与”变换后,黑游程直方图分布明显变尖,并且变换后在横坐标为1处取得的峰值比原图在横坐标为2处取得的峰值更大.峰值越大,用直方图对方法在横坐标为1处嵌入数据的容量就会越大;直方图分布越尖,直方图对方法在嵌入数据时改变的像素就越少,那么嵌入数据后图像的视觉效果就会越好.

“异或”运算:处理方法是奇偶行对应的二进制串做“异或”运算,将数据嵌在“异或”运算结果之中,相当于将数据嵌在奇偶行的变化之中.由于图像的连续性和相关性,那么奇偶行间像素的变化就比较少,相同的像素就比较多.游程编码后,黑游程的直方图统计就会向横坐标为0的方向偏移,导致横

坐标为0附近的直方图分布比较尖,有利于直方图方法的使用,嵌入的效果也更佳.图8,9分别为狒狒普通二值图像和半色调二值图像在“异或”变换前后的直方图分布.狒狒普通二值图像在进行“异或”变换之后, $x=1$ 的黑游程达到了12 704个,远远大于变换之前的5 448个;而在直方图对的整体分布方面,图8b明显比图8a更尖.狒狒的半色调图像虽然在“异或”变换以后峰值有所减小,但是直方图的整体分布变尖了,因此在嵌入同样容量的数据时,经过“异或”变换后的嵌入效果要好于变换前的效果(移位更少,变化的像素也更少).

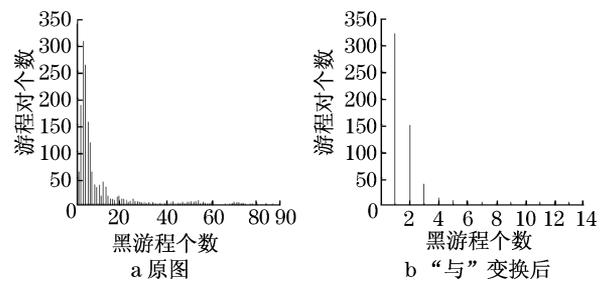


图7 米老鼠黑游程直方图

Fig.7 Black RL histogram of Mickey

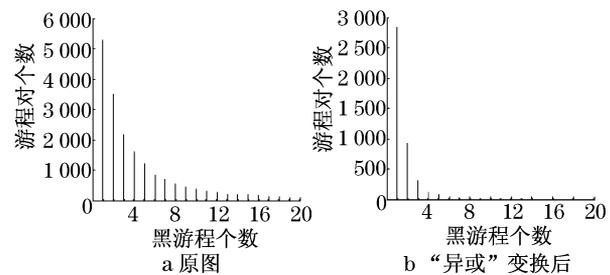


图8 狒狒普通二值图像黑游程直方图

Fig.8 Black RL histogram of non-halftone Baboon

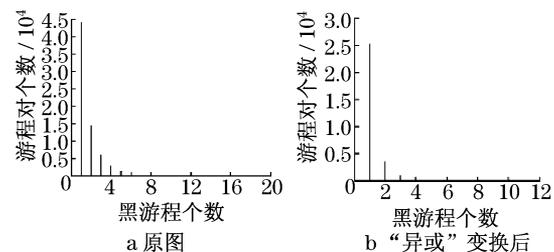


图9 狒狒半色调二值图像黑游程直方图

Fig.9 Black RL histogram of halftone Baboon

## 4 结论

提出通过逻辑运算预处理的方法,对原有的基

于直方图对的无损数据隐藏方法进行改进.

与文献[1]相比,嵌入相同容量的数据,本文的方法引起的像素改变明显要少;与文献[3]相比,对于非半色调二值图像,本文方法的嵌入容量更大,能够达到 6 532 bit,而文献[3]只有 2 572 bit;本文的方法既适用于半色调图像,又适用于非半色调图像,并且都有较好的效果,而文献[2]的方法只对半色调图像有较好的效果.综上所述,“与”运算能够提高简单纹理二值图像的嵌入效果,而“异或”运算则对复杂纹理二值图像有更突出的效果.本文的方法将数据嵌入“与”运算和“异或”运算的结果中,不但增大了嵌入容量,而且对图像的视觉效果影响较小.

#### 参考文献:

- [1] Pamboukian S V D, Kim H Y. Reversible data hiding and reversible authentication water marking for binary images [EB/OL]. [2008 - 03 - 17]. <http://www.lps.usp.br/~hae/sbseg2006-rdte.pdf>.
- [2] Pan J S, Luo H, Lu Z M. A lossless watermarking scheme for halftone image authentication [J] // International Journal of Computer Science and Network Security, 2006, 6(2): 147.
- [3] Wang C C, Chang C C, Zhang X, et al. Senary Huffman compression—a reversible data hiding scheme for binary images [C] // International Workshop on Multimedia Content Analysis and Mining (MCAM07). Weihai: [s. n.], 2007: 351 - 360.
- [4] 徐德智, 童学锋, 宣国荣, 等. 基于直方图调整的二值图像无损数据隐藏[J]. 计算机应用, 2009, 29(6): 1651.
- XU Dezhi, TONG Xuefeng, XUAN Guorong, et al. Reversible binary image data hiding by histogram modification [J]. Journal of Computer Applications, 2009, 29(6): 1651.
- [5] Xuan G R, Shi Y Q, Chai P Q, et al. Optimum histogram pair based image lossless data embedding [C] // International Workshop on Digital Watermarking (IWDW 2007). Guangzhou: [s. n.], 2007: 264 - 278.
- [6] Xuan G R, Shi Y Q, Chai P Q, et al. Reversible binary image data hiding by run-length histogram modification [C] // The 19th International Conference on Pattern Recognition (ICPR 2008). Tampa: [s. n.], 2008: 1 - 4.
- [1] Pamboukian S V D, Kim H Y. Reversible data hiding and reversible authentication water marking for binary images [EB/OL]. [2008 - 03 - 17]. <http://www.lps.usp.br/~hae/sbseg2006-rdte.pdf>.
- [4] Vamvakaki V, Fournier D, Chaniakakis N A. Fluorescence detection of enzymatic activity within a liposome based nanobiosensor [J]. Biosens Bioelectron, 2005, 21(2): 384.
- [5] Vander K, Veenendaal H R, Scheffer J H, et al. Biofilm formation and multiplication of Legionella in a model warm water system with pipes of copper, stainless steel and cross-linked polyethylene [J]. Wat Res, 2005, 39(13): 2789.
- [6] Massa S, Caruso M, Trovatielli F, et al. Comparison of plate count agar and R2A medium for enumeration of heterotrophic bacteria in natural mineral water [J]. World J Microbiol Biotechnol, 1998, 14(5): 727.
- [7] Boivin M Y, Massieux B, Breure A M, et al. Functional recovery of biofilm bacterial communities after copper exposure [J]. Environ Pollut, 2006, 140(2): 239.
- [8] Orgaz B, Kives J, Pedregosa A M, et al. Bacterial biofilm removal using fungal enzymes [J]. Enzyme Microbiol Tech, 2006, 40(1): 51.
- [9] Oulahal N, Martial-Gros A, Bonneau M, et al. Removal of meat biofilms from surfaces by ultrasounds combined with enzymes and/or a chelating agent [J]. Innov Food Sci Emerg Tech, 2007, 8(2): 192.
- [10] Yang X, Beyenal H, Harkin G, et al. Quantifying biofilm structure using image analysis [J]. J Microbiol Meth, 2000, 39(2): 109.
- [11] Dubey V, Mishra D, Jain N K. Melatonin loaded ethanolic liposomes: physicochemical characterization and enhanced transdermal delivery [J]. Eur J Pharm Biopharm, 2007, 67(2): 398.
- [12] Chongdar S, Gunasckaran G, Kumar P. Corrosion inhibition of mild steel by aerobic biofilm [J]. Electrochim Acta, 2005, 50(24): 4655.

#### (上接第 93 页)