

信息隐藏实验七 奇偶校验位隐藏法 范例

2023 年 4 月 11 日

摘要

根据实验要求：
1、隐藏：利用奇偶校验位隐藏法，实现将秘密图像嵌入到位图中；
2、提取：将秘密图像提取出来。

目录

1	实验原理	1
1.1	方法一	1
1.1.1	嵌入	1
1.1.2	提取	1
1.2	方法二	1
1.2.1	嵌入	2
1.2.2	提取	2
2	代码展示	2
3	实验结果	4

1 实验原理

利用奇偶校验位主要有两种方法：

1.1 方法一

该方法特点是翻转最低位，影响不大。

把载体划分成几个不相重叠的区域，在一个载体区域中存储一比特信息。

1.1.1 嵌入

选择 $L(m)$ 个不相重叠区域，计算出每一区域 l 的所有最低比特的奇偶校验位（即“1”的个数奇偶性）， $b_i (i = 1, 2, \dots, n)$ 。

$$b_i = \sum_{j \in I} LSB(c_j) \bmod 2$$

嵌入信息时，在对应区域的奇偶校验位上嵌入信息比特 m_i ，如果奇偶校验位 b_i 与 m_i 不匹配，则将该区域中所有元素的最低比特位进行翻转，使得奇偶校验位与 m_i 相同，即 $b_i = m_i$ 。

例如一个区域内所有像素的最低比特有偶数个 1，计算得奇偶校验位 $b_i = 0$ 。如果要嵌入的秘密信息比特为 1，即 $m_i = 1$ ，要想满足 $b_i = m_i$ 则需要翻转所有像素的最低比特位，使得该区域的最低有效位有奇数个 1，即 $b_i = 1$ ，从而满足 $b_i = m_i$ 。

1.1.2 提取

在接收端，收方与发方拥有共同的伪装密钥作为种子，可以伪随机地构造载体区域。收方从载体区域中计算出奇偶校验位，排列起来就可以重构秘密信息。

1.2 方法二

该方法特点是翻转像素少。

把载体划分成几个不相重叠的区域，在一个载体区域中存储一比特信息。

1.2.1 嵌入

选择 $L(m)$ 个不相重叠区域 I 的所有最低比特的奇偶校验位 $b_i (i = 1, 2, \dots, n)$ 。

$$b_i = \sum_{j \in I} LSB(c_j) \bmod 2$$

区域 I 隐藏一个信息比特。若 b_i 与 m_i 不同，则将该区域中某个像素的最低比特位进行翻转，使得奇偶校验位与 m_i 相同，即 $b_i = m_i$ 。

例如一个区域内所有像素的最低比特位有偶数个 1，计算得奇偶校验位 $b_i = 0$ 。如果要嵌入的秘密信息比特为 1，即 $m_i = 1$ ，要想满足 $b_i = m_i$ 则需要翻转某个像素的最低比特位，使得该区域的最低有效位有奇数个 1，即 $b_i = 1$ ，从而满足 $b_i = m_i$ 。

1.2.2 提取

用同样的方法划分载体区域，计算出奇偶校验位，构成秘密信息。

2 代码展示

本实验采用了第二种部分翻转法：

奇偶校验位函数代码如下：

```
1 function out = checksum(x,i,j)
2 %计算特定一维向量的第m个区域的最低位的校验和
3     temp=zeros(1,4);
4     temp(1)=bitget(x(2*i-1,2*j-1),1);
5     temp(2)=bitget(x(2*i-1,2*j),1);
6     temp(3)=bitget(x(2*i,2*j-1),1);
7     temp(4)=bitget(x(2*i,2*j),1);
8     out=rem(sum(temp),2);
9 end
```

加密：

这里借助随机数对 2×2 方块内的任意一个像素的最低比特进行翻转。

```

1 x=imread('lena_gray.bmp');
2 y=double(rgb2gray(imread('hide.bmp')));
3 [m,n]=size(y);
4 origin=x;
5
6 %加密
7
8 for i=1:m
9     for j=1:n
10         if checksum(x,i,j)~=y(i,j)
11             random=int8(rand()*3);
12             switch random
13                 case 0
14                     x(2*i-1,2*j-1)=bitset(x(2*i-1,2*j-1),1,~bitget(x(2*i-1,2*j-1),1));
15                 case 1
16                     x(2*i-1,2*j)=bitset(x(2*i-1,2*j),1,~bitget(x(2*i-1,2*j),1));
17                 case 2
18                     x(2*i,2*j-1)=bitset(x(2*i,2*j-1),1,~bitget(x(2*i,2*j-1),1));
19                 case 3
20                     x(2*i,2*j)=bitset(x(2*i,2*j),1,~bitget(x(2*i,2*j),1));
21             end
22         end
23     end
24 end
25 imwrite(x,'watermarkedImage.bmp');
26 subplot(2,2,1);imshow(origin,[]);title('原始图片');
27 subplot(2,2,2);imshow(y,[]);title('水印');
28 subplot(2,2,3);imshow(x,[]);title('密图');

```

解密:

```

1 c=imread('watermarkedImage.bmp');
2 [m,n]=size(c);
3 secret=zeros(m/2,n/2);
4 for i=1:m/2
5     for j=1:n/2
6         secret(i,j)=checksum(c,i,j);
7     end
8 end
9 %解密
10
11 imshow(secret,[]);
12 title('提取出的水印');

```

3 实验结果

