南開大學
Nankai University

网络空间安全学院

信息隐藏实验报告报告

奇偶校验位隐藏法

姓名：魏伯繁

学号：2011395

专业：信息安全

2023 年 4 月 17 日

目录

1 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。

2 实验原理

2.1 奇偶校验位

基于奇偶校验位的信息隐藏法是一种在数字信号传输中实现信息隐藏的方法。在信号传输中采用奇偶校验位的目的是为了错误检测和纠正。而基于奇偶校验位的信息隐藏法利用了这一特性，将隐藏信息编码到奇偶校验位中，以达到信息隐藏的目的。

具体来说，该方法的实现分为两个步骤：编码和解码。编码步骤中，将待隐藏的信息嵌入到码元的奇偶校验位中。假设原始信号采用二进制编码方式，每个码元由若干个二进制位组成。在这些二进制位中，我们选定某些位作为奇偶校验位，并对这些位进行奇偶校验。对于数据位和隐藏位，我们不进行奇偶校验，而对于奇偶校验位，我们根据数据位和隐藏位的值来指定是偶校验还是奇校验。如果数据位和隐藏位中二进制位为 1 的个数是偶数，则奇偶校验位为偶校验，反之则为奇校验。这样，将待隐藏的信息以一种巧妙的方式嵌入到码元的奇偶校验位中，而且不会引起原始信号的明显变化。

解码步骤中，首先需要提取奇偶校验位，并进行校验。如果校验结果与原始信号相同，则说明隐藏信息没有被篡改。然后，从奇偶校验位中提取出隐藏信息。对于它们的二进制形式，我们选择一种特殊编码方式，以保证在解码时可以准确地还原出原始信息。

基于奇偶校验位的信息隐藏法具有隐蔽性高、嵌入能力强等优点，但受到了噪声和攻击的影响。因此，在应用该方法时，需要考虑信号的可靠性和安全性，并结合其他的信息隐藏方法进行综合使用。

2.2 方法一: 全范围翻转

该方法特点是翻转一个区域内的所有最低位，因为一个像素的最低比特位基本是一些噪声，所以对图片观感影响不大。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。选择 L(m) 个不相重叠区域，计算出每一区域 I 的所有最低比特的奇偶校验位（即 “1” 的个数奇偶性），bi(i = 1, 2, ..., n)。

$$b_i = \sum_{j \in I}^{n} LSB(c_j) mod 2$$

也就是说，在一个区域计奇偶校验位，如果与所隐藏的信息比特位一致则说明也不做，如果与所隐藏的信息比特位不一致就将该区域的所有像素的最低比特位取反，这样就可以达到奇偶校验位也取反的目的。

2.2.1 方法二：区域内随机翻转

该方法特点是和方法一相比翻转像素少。他的原理一样是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。选择 L(m) 个不相重叠区域，计算出每一区域 I 的所有最低比特的奇偶校验位 bi(i = 1, 2, ..., n)。

$$b_i = \sum_{j \in I}^{n} LSB(c_j) mod 2$$

也就是说，在一个区域计奇偶校验位，如果与所隐藏的信息比特位一致则说明也不做，如果与所隐藏的信息比特位不一致就随机挑选该区域的一个像素，将该像素的最低比特位进行翻转，这样也能够达到翻转奇偶校验位的目的。

3 编程实现

3.1 计算校验和

这段代码是用来计算一个特定二维矩阵 x 的第 i 行第 j 列对应的四个区域 (2i-1,2j-1)、(2i-1,2j)、(2i,2j-1)、(2i,2j) 的最低位的校验和。

具体地，这个二维矩阵 x 可以看做是一张图片的分块内容，而每个分块中又被分为 4 个区域，每个区域中又被编码为 8 个二进制数。在这个编码方式中，每个区域中的 8 个二进制数最低位是校验位，可以用来检测这个区域中是否存在一位出错。因此，通过计算这 4 个区域中最低位的校验和，就可以检测出这个分块中是否存在偶数个位出错。这个检测方式被用在很多图片传输和存储的应用中，能够增加数据传输的可靠性和容错性。

具体而言，该函数会读取输入矩阵 x 中的 (2i-1,2j-1)、(2i-1,2j)、(2i,2j-1)、(2i,2j) 四个像素的最低位，并将这些最低位的值存储在一个长度为 4 的向量 temp 中。然后将 temp 中的所有元素求和，并用 2 取余，得到这四个区域最低位的校验和 out。

```
    % 计算特定一维向量的第 m 个区域的最低位的校验和
    temp= zeros(1, 4);
    temp(1) = bitget(x(2*index_row-1,2*index_col-1), 1);
    temp(2) = bitget(x(2*index_row-1,2*index_col), 1);
    temp(3) = bitget(x(2*index_row, 2*index_col-1), 1);
    temp(4) = bitget(x(2*index_row, 2*index_col ), 1);
    out=rem(sum(temp), 2);
```

3.2 信息隐藏

这段代码的主要功能是利用校验和技术实现数字水印的嵌入和提取。数字水印是一种在数字图像、音频、视频等各种媒体中隐藏特定信息的技术，其可以在保证原始媒体质量的基础上，为媒体提供版权保护、信息隐藏、身份认证等功能，应用广泛。

在这个代码中，输入变量 x 表示原始图像，y 是对该图像进行处理后得到的校验和矩阵，row_size 和 col_size 分别是原始图像的行和列数。在第一个嵌入的步骤中，代码首先使用 checksum 函数计算 x 矩阵中每个 2*2 区域的最低位校验和，并与 y 矩阵中对应位置的值比较，如果不一致则需要嵌入数字水印。

接着，程序将生成一个随机数（0-2 之间的一个整数），然后在当前 2*2 的区域中随意选取一位进行反转（即将当前位的值取反）。反转的位置由生成的随机数决定，每个数字代表 4 个角中的一个，其中 0 表示左上角，1 表示右上角，2 表示左下角，3 表示右下角。

用这种方式嵌入数字水印可以实现比较良好的鲁棒性，即使在一定程度的攻击下，数字水印也能保持稳定。最后，程序会将嵌入了水印的图像保存到另一个文件中，存储为 bmp 格式文件，同时在结果中返回嵌入了水印的图像。利用嵌入和提取两段代码，我们就可以实现数字水印技术的应用实现。

```
for index_row =1:row_size
    for index_col =1:col_size
        if checksum(x, index_row, index_col) ~= y(index_row, index_col) % 需要反转一位
            random= int8(rand()*3);
            switch random  % 任意反转一位
                case 0
                    x(2*index_row-1,2*index_col-1)= bitset(x(2*index_row-1,2*index_col-1),
                    1, ~ bitget(x(2*index_row-1,2*index_col-1), 1));
                case 1
                    x(2*index_row-1,2*index_col)= bitset(x(2*index_row-1,2*index_col) , 1
                    , ~ bitget(x(2*index_row-1,2*index_col), 1));
                case 2
                    x(2*index_row, 2*index_col-1)= bitset(x(2*index_row, 2*index_col-1) ,1
                    ,~ bitget(x(2*index_row , 2*index_col-1) , 1));
                case 3
                    x(2*index_row , 2*index_col)= bitset(x(2*index_row , 2*index_col) , 1
                    , ~ bitget(x(2*index_row , 2*index_col) , 1));
            end
        end
    end
end
imwrite(x , 'watermarkedImage.bmp');
result=x;
```

3.3 信息提取

这段代码实现的是数字水印的提取功能。与数字水印的嵌入相比，数字水印的提取成为了一项更加关键的技术，利用提取功能，我们可以获取到嵌入在图像中的数字水印，判断其是否被篡改，从而保证信息的真实性和完整性。

这段代码的第一行使用 imread 函数从文件中读取图像并将其加载到内存中。接着，程序使用 size 函数计算图像的行和列数，其中 row_size 表示图像高度，col_size 表示图像宽度。

随后，程序创建一个以 secret 命名的矩阵，并将其初始化为值为 0 的矩阵。该矩阵用于存储提取出的数字水印信息。

在接下来的两个嵌套循环中，程序对每个 2*2 的区域进行校验和计算，并将结果存储到 secret 矩阵中。具体来说，程序通过 checksum 函数计算 c 矩阵中的每个 2*2 区域的校验和，并将计算结果存储到 secret 矩阵中对应位置的单元格中。由于之前嵌入时会将校验和数字水印存储到照片中，且存储时将原始照片分成了 2x2 的小块，因此我们在提取时同样需要对照片采用 2x2 的块作为计算单位。

最后，out 变量存储 secret 矩阵，并在结果中返回提取出的数字水印信息。由此，我们可以获取到嵌入在图像中的数字水印信息，从而判断其是否被篡改，保证图像信息的安全。

```
c=imread('watermarkedImage.bmp');
```

```
[row_size, col_size]= size(c);
secret = zeros(row_size/2 , col_size/2);
for index_row =1:row_size/2
    for index_col =1: col_size/2
        secret(index_row, index_col)= checksum(c, index_row, index_col);
    end
end
out=secret;
```

最终的实验效果如下图所示：

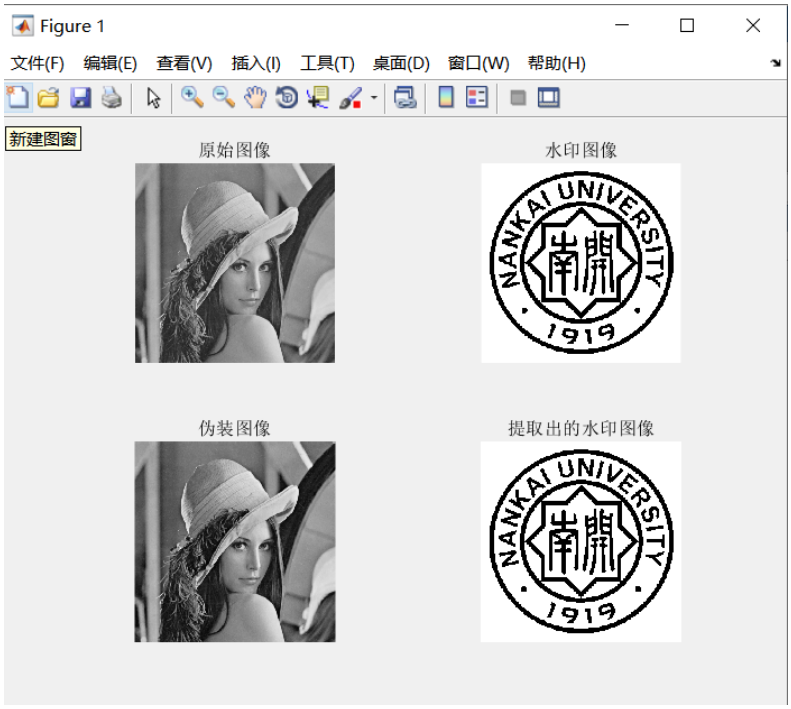

图 3.1: 实验效果图

4 实验总结

基于奇偶校验位的信息隐藏技术可以在数字图像、音频、视频等媒体中隐藏特定信息，是数字水印技术最基本的形式之一。该技术的基本原理是在数字媒体中嵌入带有特定意义的校验位，使得校验位的嵌入不会对原始图像的视觉效果造成太大影响，从而实现隐蔽性。在数据传输过程中，接收端通过重新计算校验位和初始的校验位进行比对，以判断是否有数据被篡改或修改，实现了数字媒体信息的完整性验证，达到保护信息的目的。

奇偶校验位的信息隐藏方法是其中较为常见、基本的一种方式。奇偶校验首先将数据按一定方法分组，将几位作为一组，通过计算该组数据中 1 的个数，来决定一位校验位的取值。如果 1 的个数为偶数，则校验位就是 0，否则校验位就是 1。在信息隐藏过程中，则是将需要嵌入的信息转变为一定的二进制码，再嵌入进原数据的校验位当中，利用嵌入时会修改或反转一定的校验位数来嵌入信息，通过这种方式实现隐蔽性和鲁棒性。

与其他疏水和扩频等数字水印技术相比，基于奇偶校验位的嵌入提取算法较为简单，因此在一些

安全性要求不是很高的应用场景中更加常见。然而，奇偶校验位的信息隐藏容量有限，只适用于需要低密度数据嵌入的应用场景。同时，奇偶校验位也不具有抗 JPEG 格式噪声的优势，因此在网络传输和存储过程中要注意防止由于数据压缩等操作导致的信息丢失。

总之，基于奇偶校验位实现的信息隐藏技术是数字水印技术中最基础的形式之一，适用于一些较为简单的信息隐藏和验证应用中。但需要注意的是，若要在实际应用中使用该技术，需要注意隐蔽性和鲁棒性的平衡。同时要注意根据数据嵌入的需要来选择不同的数字水印技术，以保证实际应用效果的稳定和满足需求。