

华中科技大学

# 课程实验报告

课程名称： 多媒体数据安全实验

专业班级： 信息安全 201802

学 号： U201812058

姓 名： 程启涵

指导教师： 马晓静

报告日期： 2021 年 4 月 12 日

网络空间安全学院

## 目录

1 基于 LSB 的空域信息隐藏实现.....	3
1.1 问题描述.....	3
1.2 系统设计.....	3
1.2.1 嵌入算法设计.....	3
1.2.2 提取算法设计.....	4
1.3 系统实现.....	4
1.3.1 嵌入信息实验流程.....	4
1.3.2 提取信息流程.....	5
1.4 实验小结.....	6
2 JSTEG 图像变换域信息隐藏实现.....	7
2.1 问题描述.....	7
2.2 系统设计.....	7
2.2.1 嵌入信息算法设计.....	7
2.2.2 提取算法设计.....	7
2.3 系统实现.....	8
2.3.1 嵌入信息流程.....	8
2.3.2 提取信息流程.....	10
2.4 实验小结.....	11
3 F3 图像变换域信息隐藏实现.....	12
3.1 问题描述.....	12
3.2 系统设计.....	12
3.2.1 嵌入信息算法设计.....	12
3.2.2 提取信息算法设计.....	12
3.3 系统实现.....	12
3.3.1 嵌入信息算法设计.....	12
3.3.2 提取信息算法设计.....	12
3.4 实验总结.....	16
4 F4 图像变换域信息隐藏实现.....	17
4.1 问题描述.....	17
4.2 系统设计.....	17
4.2.1 嵌入信息算法设计.....	17
4.2.2 提取信息算法设计.....	17
4.3 系统实现.....	18
4.3.1 嵌入信息流程.....	18
4.4.2 嵌入信息提取.....	20
4.4 实验总结.....	21
5 F5 图像变换域信息隐藏实现.....	22
5.1 问题描述.....	22
5.2 系统设计.....	23
5.2.1 嵌入信息算法设计.....	23
5.2.2 提取信息算法设计.....	23
5.3 系统实现.....	23

5.3.1 嵌入信息流程.....	24
5.3.2 嵌入信息提取.....	24
5.4 实验总结.....	25
6 实验总结感想.....	26
参考文献.....	27

# 1 基于 LSB 的空域信息隐藏实现

## 1.1 问题描述

通过实验达到(1)加深对空域信息隐藏概念、系统的理解；(2)熟悉数字图片格式；(3)掌握 MATLAB 基础操作。

### 1.1.1 LSB 空域信息隐藏算法

LSB空域信息隐藏算法关键步骤是将原始图像最低一个位平面替换为要隐藏的秘密信息。首先要将原始图像的最低位面设置成 0，为此可以判断像素点的灰度值是否是偶数，如果不是偶数，就将最低位取反。之后就可以一位位地根据秘密信息改变像素地最低有效位，实现秘密信息地嵌入。

## 1.2 系统设计

### 1.2.1 嵌入算法设计

- 对原始图像中地每个像素点地灰度值进行变换，从十进制转换成二进制
- 将秘密信息提取成二进制序列，并将原始图像中像素点地最低比特位替换。  
以替换信息 1101011110101101 为例，如图 1-1-1 所示

5	11	10	13	
20	31	41	51	
27	10	17	44	
37	85	14	35	

Stego Image

图 1-1-0 替换前图像地矩阵信息

Least Significant Bit (LSB)

0000010 <b>1</b>	0000101 <b>0</b>	0000101 <b>0</b>	0000110 <b>0</b>
0001010 <b>0</b>	0001111 <b>1</b>	0010100 <b>0</b>	0011001 <b>1</b>
0001101 <b>1</b>	0000101 <b>0</b>	0001000 <b>1</b>	0010110 <b>0</b>
0010010 <b>1</b>	0101010 <b>1</b>	0000111 <b>0</b>	0010001 <b>0</b>

图 1-1-1 嵌入信息后的图像矩阵

- 在上述替换结束后，将像素点的二进制数据转换回十进制数据，保存。

## 1.2.2 提取算法设计

- 对载密图像中的每个像素点的灰度值进行变换，从十进制转换成二进制
- 提取图像像素点中最低有效位的数据，根据嵌入顺序进行组合得到原秘密序列
- 将原秘密序列的二进制数据写入文件中

## 1.3 系统实现

### 1.3.1 嵌入信息实验流程

嵌入信息实验流程如图 1.3.1 所示



图 1-3-1 嵌入信息流程

在运行完嵌入实验后，得到如图 1-3-2 所示的原始图像和载密图像，可以看出两幅图像在视觉上不可分辨。

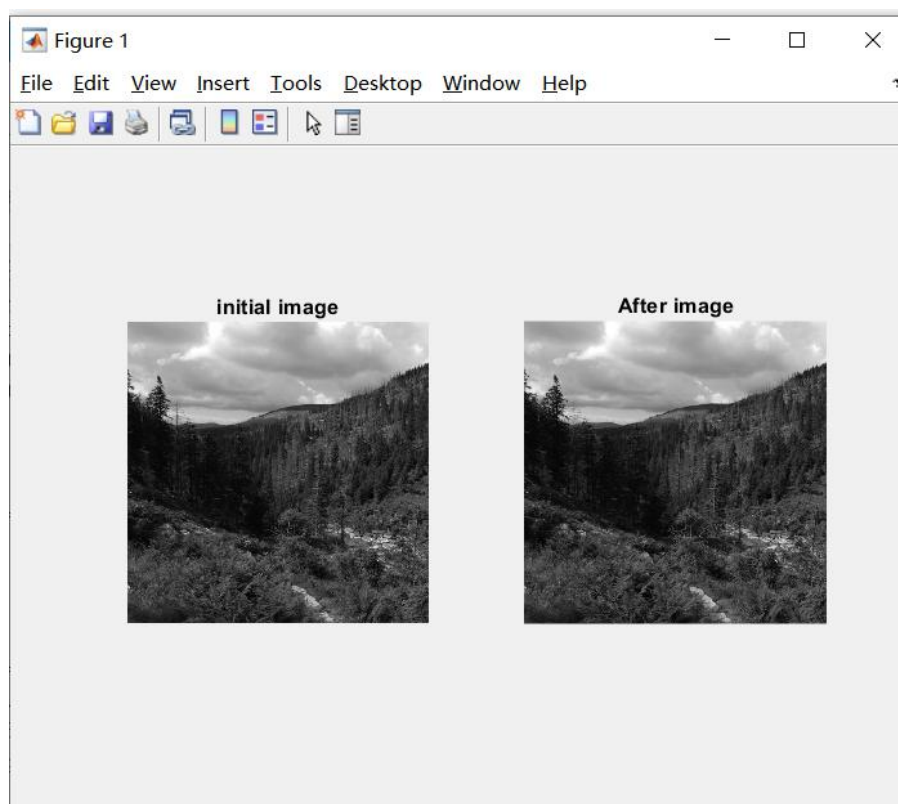


图 1-3-2 原始图像和载密图像

### 1.3.2 提取信息流程

提取信息流程如图 1-3-2 所示

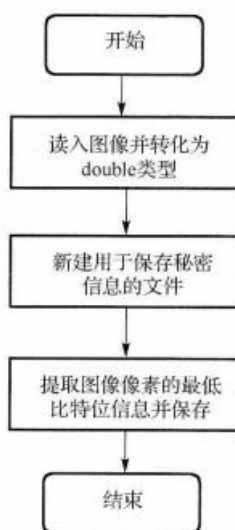


图 1-3-2 嵌入信息提取流程

简单说一下如何提取图像像素的最低位比特位信息，如下图 1-3-3 所示

```
if bitand(Picture(f1,f2),1)==1
    fwrite(frr,1,'ubit1');
    result(p,1)=1;
else
    fwrite(frr,0,'ubit1');
    result(p,1)=0;
```

图 1-3-3

直接提取图像的最后一位，使用与运算得到最后一位，直接写入文件。

运行结果如下图 1-3-3 所示

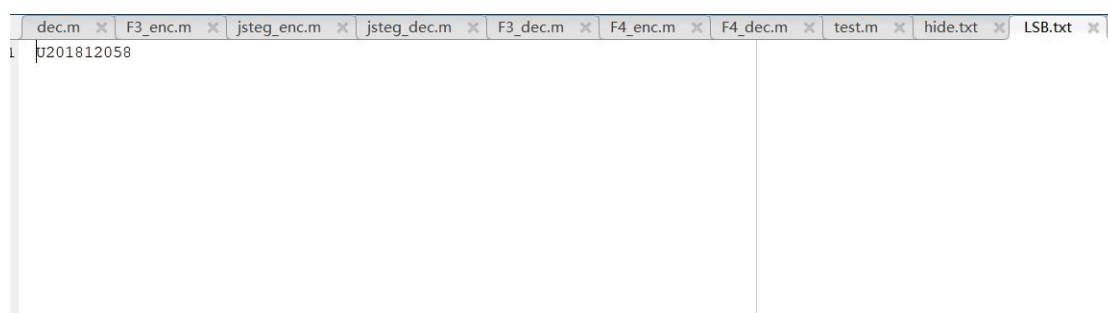


图 1-3-2 提取信息

## 1.4 实验小结

LSB 算法是按照像素点的顺序进行信息隐藏，方法比较简单，提取也只要取出最后一位就可以实现解密，安全性不够好，后续的 F3 等算法利用随机隐写可以提高信息隐藏的安全性。

信息隐写的原理建立在图像压缩的基础之上除 LSB 针对 BMP 类的图像，上课只要认真听讲，后续实验基本可以说毫无压力，一个下午就可以完成。

## 2 JSTEG 图像变换域信息隐藏实现

### 2.1 问题描述

要求实现 JSTEG 信息嵌入与提取算法。并比较嵌入前后的视觉效果与 DCT 系数直方图。

#### 2.1.1 JSTEG 隐密算法

JSTEG 是最早以 JPEG 图像为载体的隐秘算法，主要是利用 LSB 替换思想在 DCT 域实现。主要思路是：将一个二进制的隐密信息，嵌入到量化后的 DCT 系数的 LSB 上，但是对原始的 0，1，DCT 系数除外。提取隐密信息时，只需要将载密图像中不等于 0 和 1 的量化的 DCT 系数提出即可。

### 2.2 系统设计

#### 2.2.1 嵌入信息算法设计

- 提取原始图像中的量化后的 DCT 系数，即 DC 系数
- 将秘密信息提取成二进制序列，并将原始图像中像素点中根据 JSTEG 加密思想进行替换
- 保存图像

#### 2.2.2 提取算法设计

- 提取载密图像的 DC 系数
- 根据 JSTEG 加密流程，逆向取出嵌入的比特信息，并写入文件
- 保存提取信息



## 2.3 系统实现

### 2.3.1 嵌入信息流程

嵌入信息流程如图 2-3-1 所示



其中 JSTEG 的替换流程如下图 2-3-2 所示

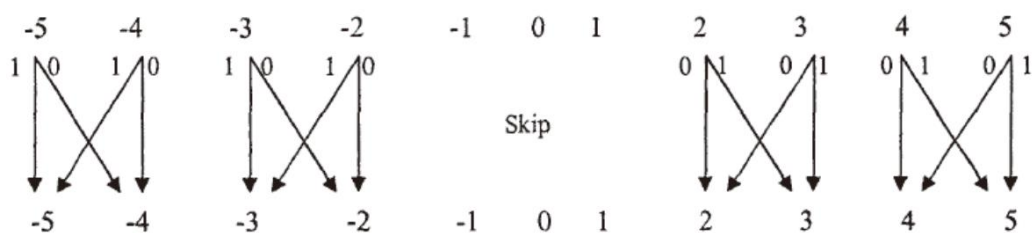


图 2-3-2

主要流程就是 DC 系数和比特流进行匹配，值时 0，1，-1 的 DC 系数就自动跳过，其他的 DC 系数对 2 取余，余数如果和比特流的数字相同就不变，负奇数遇到 0 就加一，负偶数遇到 1 就减一，正奇数遇到 0，就减一，正偶数遇到 1 就加一。

在运行完嵌入实验后，得到如图 2-3-3 所示的原始图像和载密图像，可以看出两幅图像在视觉上不可分辨。

图 2-3-3

待加密信息如下图所示

如下图 2-3-4 所示, 出现了明显的成对效应。

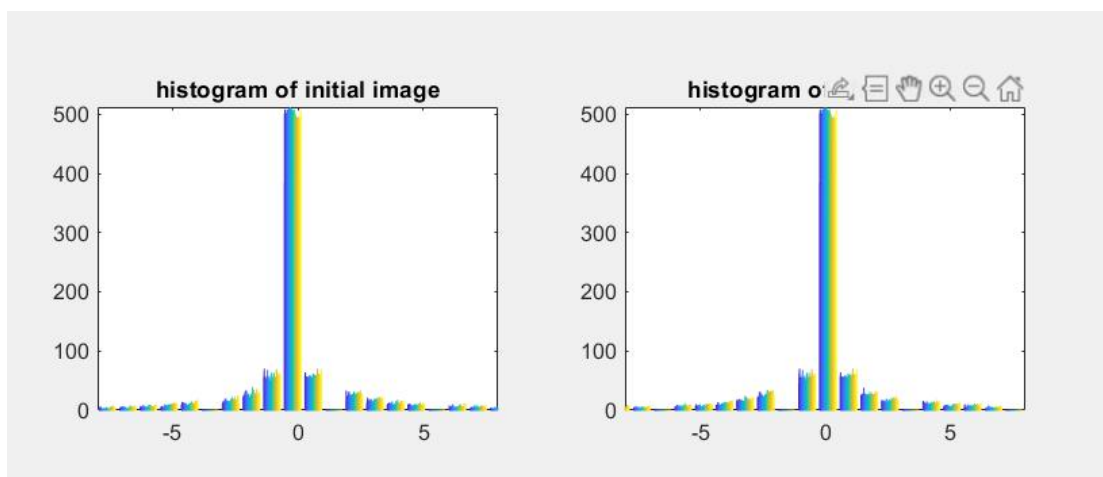


图 2-3-4

JPEG 图像的 DCT 系数通常满足以下三个特性

- (1) DCT 系数的绝对值越大，其对应直方图中的值出现频率就越小。
- (2) 随着 DCT 系数绝对值的增大，其出现频率下降的幅度减小。
- (3) 各系数出现的频率关于 0 对称

### 2.3.2 提取信息流程

提取信息的流程如下图 2-2-3 所示



图 2-2-3

JSTEG 解密的逆向流程简单总结就是得到载密图像的 DC 系数，跳过 0，1，-1 的 DC 系数。DC 系数是正偶数，加密信息就是 0，正奇数，加密信息就是 1，负偶数加密信息就是 0，负奇数加密信息就是 1。

一个学号是 80 个比特，当我设置，提取信息比特数量是 800 时，解密信息应该时 10 个重复的学号信息，解密结果如下图 2-2-4 所示

```
AC=numel(dct)-numel(dct(1:8:end,1:8:end));
len=800;
p=1;
[m,n]=size(dct);
```

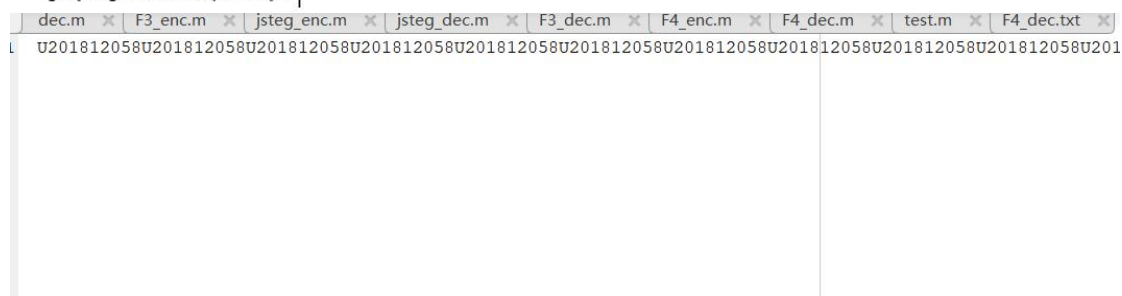


图 2-2-4

## 2.4 实验小结

老师上课讲的 JSTEG 算法的加密是一种连续的替换，虽然易实现但是安全性较差，在 LSB 实验中我就在想用随机序列来进行加密，应该也是可行的，在网上查资料果然是可以随机选择元素，生成一个伪随机数序列，调整随机序列的均值，这样就可以确保子集 S 中的元素分散在集合 C 中。接收器就可以通过同样的伪随机数种子和发生器从集合 C 重建子集 S。

信息隐写的原理建立在图像压缩的基础之上，JPEG 图像的压缩原理需要好好掌握，discrete cosine transform 非常的巧妙，虽然实现有点麻烦，但是还好老师提供了封装好的工具，实验上手应该没有难度。

## 3 F3 图像变换域信息隐藏实现

### 3.1 问题描述

要求实现 F3 信息嵌入与提取算法。并比较嵌入前后的视觉效果与 DCT 系数直方图。

#### 3.1.1 F3 隐密算法

和 JSTEG 不同的是，F3 隐秘算法只修改不为 0 的 DCT 系数，当秘密信息和 DCT 系数的最低比特位不一致时，将 DCT 的绝对值数减一。

### 3.2 系统设计

#### 3.2.1 嵌入信息算法设计

- 载体图像的每个非 0 的 DCT 系数的 LSB，用来隐藏一比特的信息，若加密信息的比特位和 DCT 系数的 LSB 相同，就不进行修改，若不同就绝对值减一，0 就跳过。
- 若原 DCT 系数为+1 或-1，而待嵌入秘密比特位为 0，则原系数会变为 0，本次嵌入操作无效，重新选择嵌入位。

#### 3.2.2 提取信息算法设计

- 将载密信息不为 0 的 DCT 系数的最低比特位按序取出即为秘密信息。

### 3.3 系统实现

#### 3.3.1 嵌入信息流程

嵌入信息实际操作流程如下图 3.3.1 所示

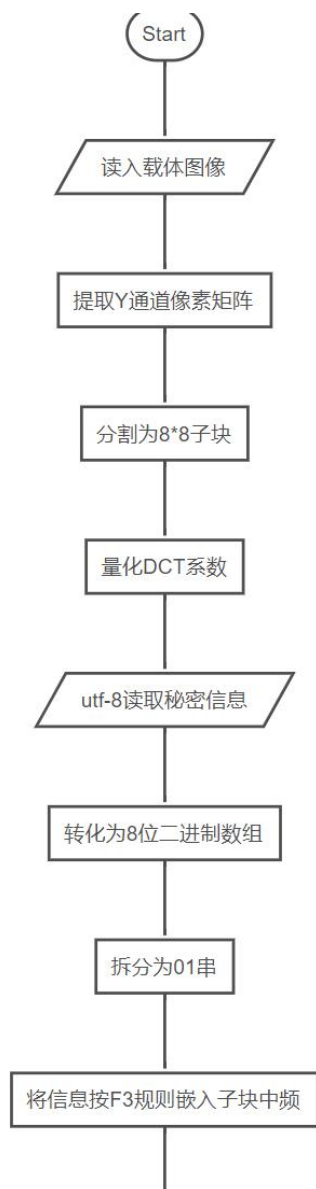


图 3.3.1

但是老师给我们提供了现成 JPEG 解码编码的工具，我们可以直接得到 DC 系数。

下图 3.3.2 是加密后的图像和加密前的图像，以及前后 DCT 系数的对比

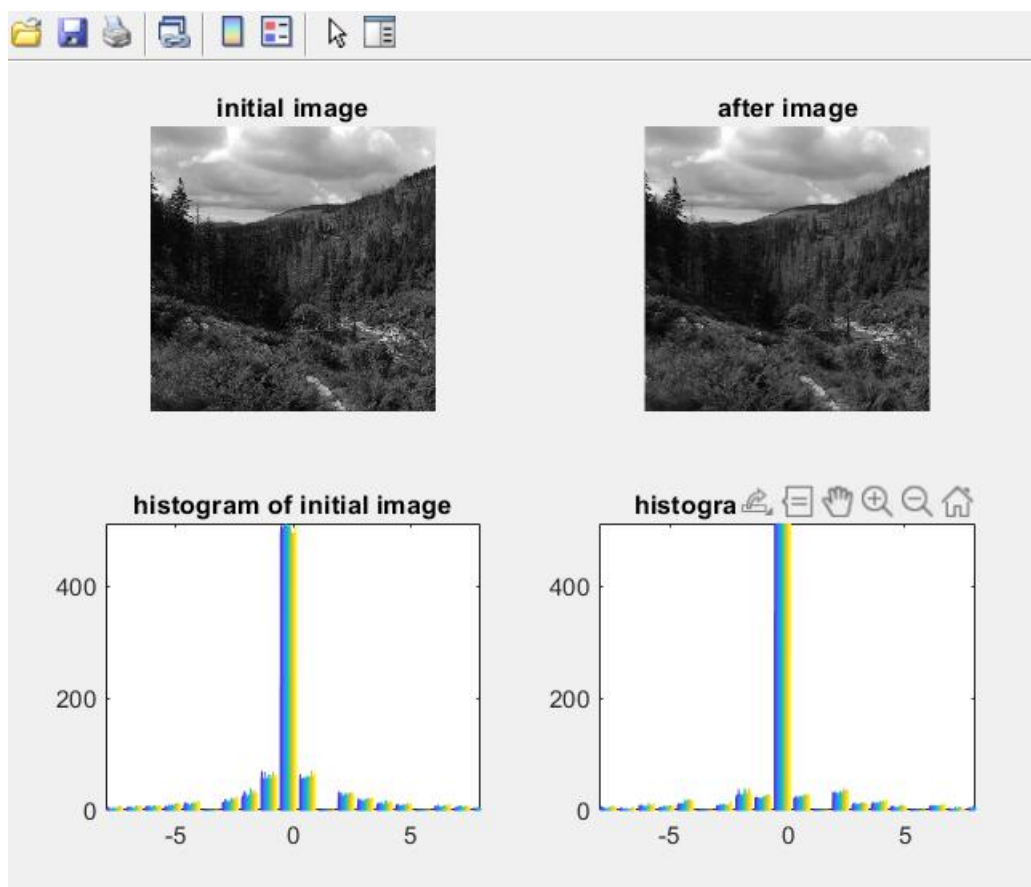


图 3.2.2

我们可以从图像看出，加密后的 DCT 系数偶数明显增加。其原因是由于无效隐藏的存在，载密图像中将会嵌入比原隐密信息更多的 0，因此得到的有差异的直方图，这一点就容易引起隐密分析者的怀疑。

### 3.3.2 提取信息流程

信息提取流程如下图 3.3.3 所示



图 3.3.3

根据 F3 加密的原理逆向推导，F3 加密的原理如下图 3.3.4 所示

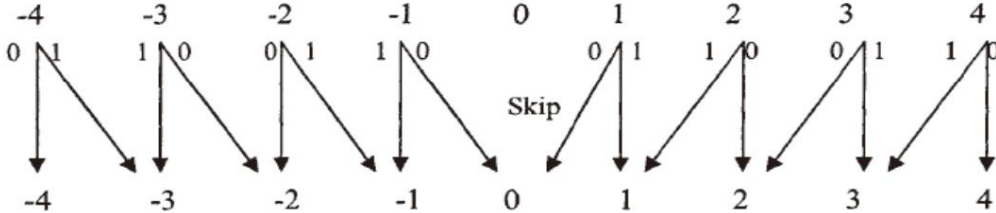


图 3.3.4

将图像中不为 0 的 DCT 系数最低比特位按序取出即可。

实验效果见下图 3.3.5 所示



图 3.3.5

### 3.4 实验总结

## 4 F4 图像变换域信息隐藏实现

### 4.1 问题描述

要求实现 F4 信息嵌入与提取算法。并比较嵌入前后的视觉效果与 DCT 系数直方图。

#### 4.1.1 F4 隐密算法

与 F3 算法不同的是，F4 算法用正奇数和负偶数来代表秘密信息 1，其他来代表秘密信息 0。

### 4.2 系统设计

#### 4.2.1 嵌入信息算法设计

- 载体图像的每个非 0 的 DCT 系数的 LSB，用来隐藏一比特的信息，若加密信息的比特位和 DCT 系数的 LSB 不同，就不进行修改，若想同就绝对值减一，0 就跳过。
- 若原 DCT 系数为+1 或-1，而待嵌入秘密比特位为 0，则原系数会变为 0，本次嵌入操作无效，重新选择嵌入位。

#### 4.2.2 提取信息算法设计

- 取出载密信息不为 0 的 DCT 系数的最低比特位。
- DCT 系数大于 0 时，取出 DCT 系数的 LSB
- DCT 系数小于 0 时，取出 DCT 系数的 LSB，LSB=0 时，嵌入比特信息为 1，LSB=1 时，嵌入信息是 0。

## 4.3 系统实现

### 4.3.1 嵌入信息流程

嵌入信息流程如下图 4.3.1 所示

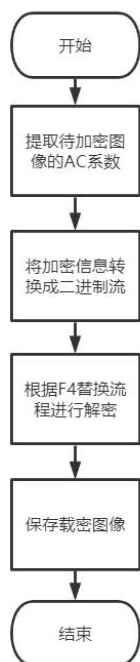


图 4.3.1

F4 的替换流程如下图 4.3.2 所示

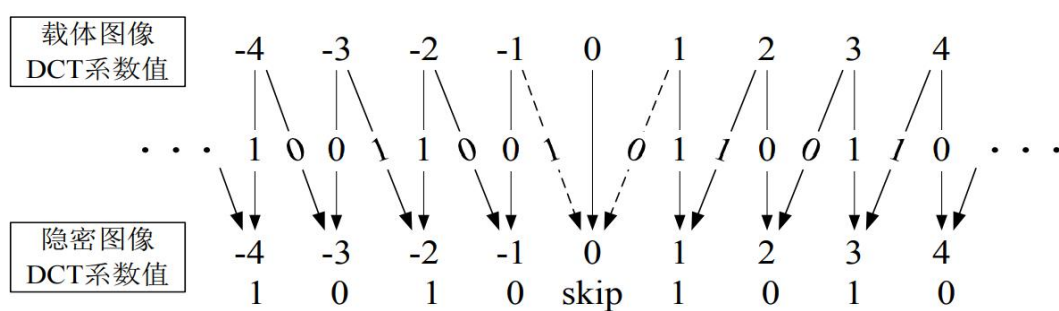


图 4.3.2

DCT 系数大于 0 时，取出 DCT 系数的 LSB，DCT 系数小于 0 时，取出 DCT 系数的 LSB，LSB=0 时，嵌入比特信息 1，LSB=1 时，嵌入信息是 0。

下图 4.3.3 是加密后的图像和加密前的图像，以及前后 DCT 系数的对比

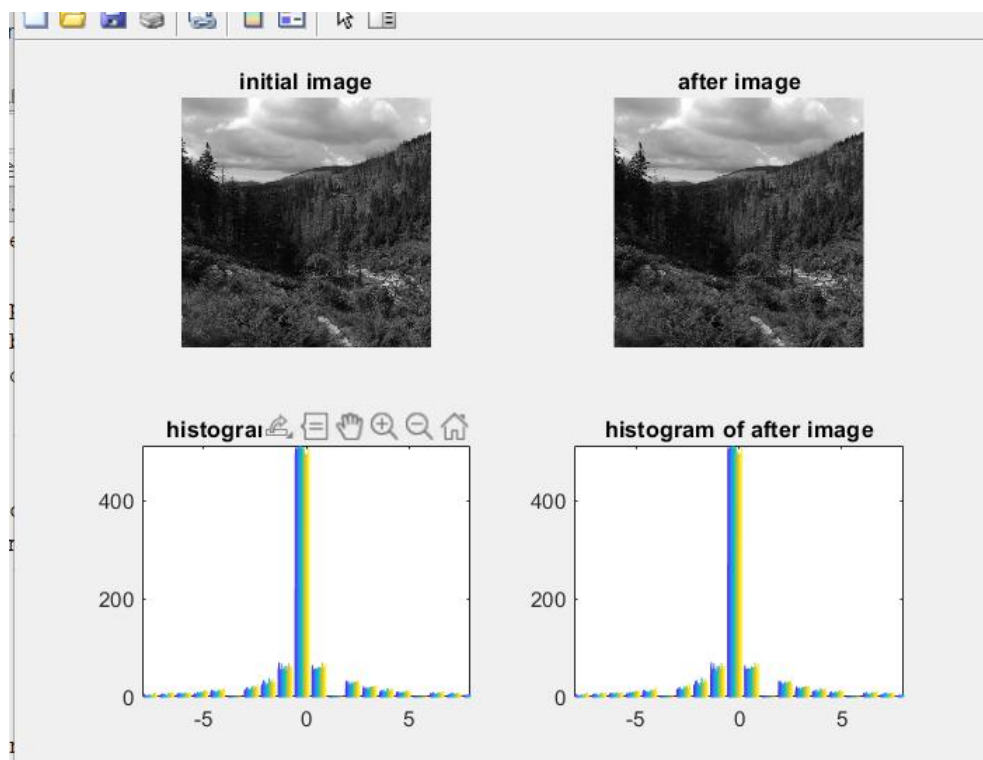


图 4.3.3

根据加密的流程我们很容易得出，不论嵌入的秘密比特是 0 还是 1，都可能产生无效隐藏而需要重新嵌入，这样的自然图像所具有的 DCT 系数直方图特性就不会被破坏，可见 F4 的安全性更高。

#### 4.4.2 嵌入信息提取

嵌入信息提取流程如下图 4.3.4 所示

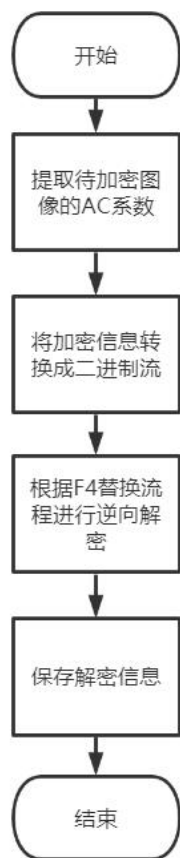


图 4.3.4

取出载密信息不为 0 的 DCT 系数的最低比特位。DCT 系数大于 0 时，取出 DCT 系数的 LSB，DCT 系数小于 0 时，取出 DCT 系数的 LSB，LSB=0 时，嵌入比特信息为 1，LSB=1 时，嵌入信息是 0。

实验效果见下图 4.3.5 所示

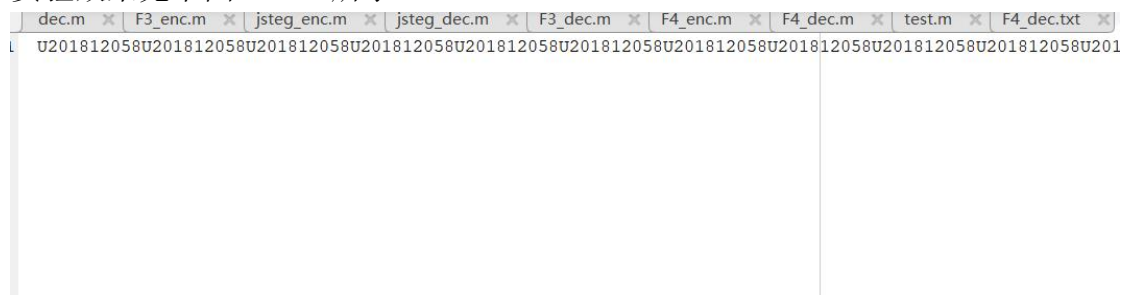


图 4.3.5

方便起见，我只提取了前 800 比特的嵌入信息。

## 4.4 实验总结

与 F3 算法不同的是，F4 算法用正奇数和负偶数来代表秘密信息 1，其他来代表秘密信息 0，论嵌入的秘密比特是 0 还是 1，都可能产生无效隐藏而需要重新嵌入，这样的自然图像所具有的 DCT 系数直方图特性就不会被破坏，可见 F4 的安全性更高。但是 F4 本身也存在缺陷，F4 是顺序嵌入，所以他的安全性不够好，假如攻击者截获图像后，虽然发现但看图像 DCT 系数本身可能没有什么差别，但是他可以使用这些加密算法解密方法，依次尝试，很容易就可以提取出关键信息。

## 5 F5 图像变换域信息隐藏实现

### 5.1 问题描述

要求实现 F5 信息嵌入与提取算法。并比较嵌入前后的视觉效果与 DCT 系数直方图。

#### 5.1.1 F5 隐密算法

F5 隐密算法是在 F4 隐密算法的基础上进行了改进。首先，F4 隐密算法是顺序嵌入，F5 改成了随机嵌入信息，其次 F5 使用了矩阵编码的思想。矩阵编码的好处就是减小了对图像的修改程度。

下面简单说明矩阵编码的原理。

例：k=2, b1, b2 是 2 个秘密信息比特，a1, a2, a3 是 3 个位置的 LSB， $\oplus$  表示异或；

b1=a1  $\oplus$  a3, b2=a2  $\oplus$  a3, 则不修改数据；

b1  $\neq$  a1  $\oplus$  a3, b2=a2  $\oplus$  a3, 则修改 a1；

b1=a1  $\oplus$  a3, b2  $\neq$  a2  $\oplus$  a3, 则修改 a2；

b1  $\neq$  a1  $\oplus$  a3, b2  $\neq$  a2  $\oplus$  a3, 则修改 a3。

提取隐密信息时，只需进行逆操作，即 b1=a1  $\oplus$  a3, b2=a2  $\oplus$  a3。

嵌入两比特的隐密信息平均只需修改 3/4 个 LSB，而普通的 LSB 隐密需要修改一个 LSB，嵌入效率提高了，而 F5 算法应用矩阵编码，目的就是为了提高 LSB 隐密算法的嵌入效率。

这样说其实还不够直观，我举一个具体的例子，

k= 3, 则 n = 7 有

a1, a2, a3, a4, a5, a6, a7, x1, x2, x3

假设：

a1 = 1; a2 = 1; a3 = 0; a4 = 1; a5 = 0; a6 = 0; a7 = 1

x1 = 1; x2 = 1; x3 = 0

f(a) = 1\*1  $\oplus$  1\*2  $\oplus$  1\*4  $\oplus$  1\*7 = 001  $\oplus$  010  $\oplus$  100  $\oplus$  111 = 000 = 0

s = f(a)  $\oplus$  x = 000  $\oplus$  110 = 110 = 6

则改变 a6 为 1 即可完成编码嵌入 x1, x2, x3

提取方法：

f(a') = 1\*1  $\oplus$  1\*2  $\oplus$  1\*4  $\oplus$  1\*6  $\oplus$  1\*7 = 001  $\oplus$  010  $\oplus$  100  $\oplus$  110  $\oplus$  111 = 110

即  $x_1, x_2, x_3$  分别为 1, 1, 0, 提取正确  
若  $s = 0$ , 则不用修改就可以实现嵌入;

## 5.2 系统设计

### 5.2.1 嵌入信息算法设计

实际的 F5 算法嵌入过程分为 5 个步骤。

- 获得输入图像的 DCT 系数。
- 用户设定的密钥作为伪随机数字产生器(PRNG)的种子, 该伪随机数字产生器用于对量化后的 DCT 系数重排列, 而且该伪随机数字生成器还要产生随机数与要嵌入的信息进行异或。
- 根据矩阵编码的参数嵌入信息, 计算确定矩阵编码参数  $k, n$ , 其中,  $n = 2k - 1$ 。
- 利用矩阵编码技术嵌入隐密信息。
  - 1: 取出待嵌入的  $k$  比特隐密信息, 使用伪随机数生成器生成的伪随机序列与  $k$  比特隐密信息进行异或运行, 得到  $K$  比特随机化 01 流, 取出  $n$  个非 0 的 DCT 系数
  - 2: 根据矩阵编码计算是否需要修改 DCT 系数, 不需要就返回 1 进行下一组嵌入, 如何过需要 DCT 系数的绝对值减一, 符号不变。
  - 3: 判断 2 中的 DCT 系数变换后是否是 0, 没有返回 1 进行下一组, 是 0, 那么无效重新选择, 返回 2 继续, 循环 123 一直到嵌入结束
- 嵌入完成后, 保存图像

### 5.2.2 提取信息算法设计

提取秘密信息的时候, 先得到图像量化后的 DCT 系数, 根据预先设定的密钥产生伪随机序列对 DCT 系数进行处理, 然后根据矩阵编码的参数提取加密信息。

## 5.3 系统实现

### 5.3.1 嵌入信息流程

嵌入信息流程如下图 5.3.1 所示





图 5.3.1

下面对代码做出一些解释，便于直观了解。

```

changeable=find(changeable);%记录非0 系数的索引
rand('state',SEED);%根据密钥种子产生伪随机数重排列量化DCT系数
changeable=changeable(randperm(AC));%随机化AC序列
idD=1;
len=length(message);
for id =1:len
    while(abs(dct(changeable(idD)))<=1)%不考虑DC系数和值为0的AC系数。<=1的目的就是-1，1处理后会变成0，嵌入无效
        dct(changeable(idD))=0;
        idD=idD+1;
        if(idD>=AC)
            break;
        end
    end
    if(message(id,1)~=mod(dct(changeable(idD)),2))%根据矩阵编码计算是否需要修改DCT系数，实际上并没有进行矩阵编码在nsf5里
        dct(changeable(idD))=dct(changeable(idD))-sign(dct(changeable(idD)));%sign函数是符号函数
    end
end
    
```

下图 5.3.3 是加密后的图像和加密前的图像，以及前后 DCT 系数的对比

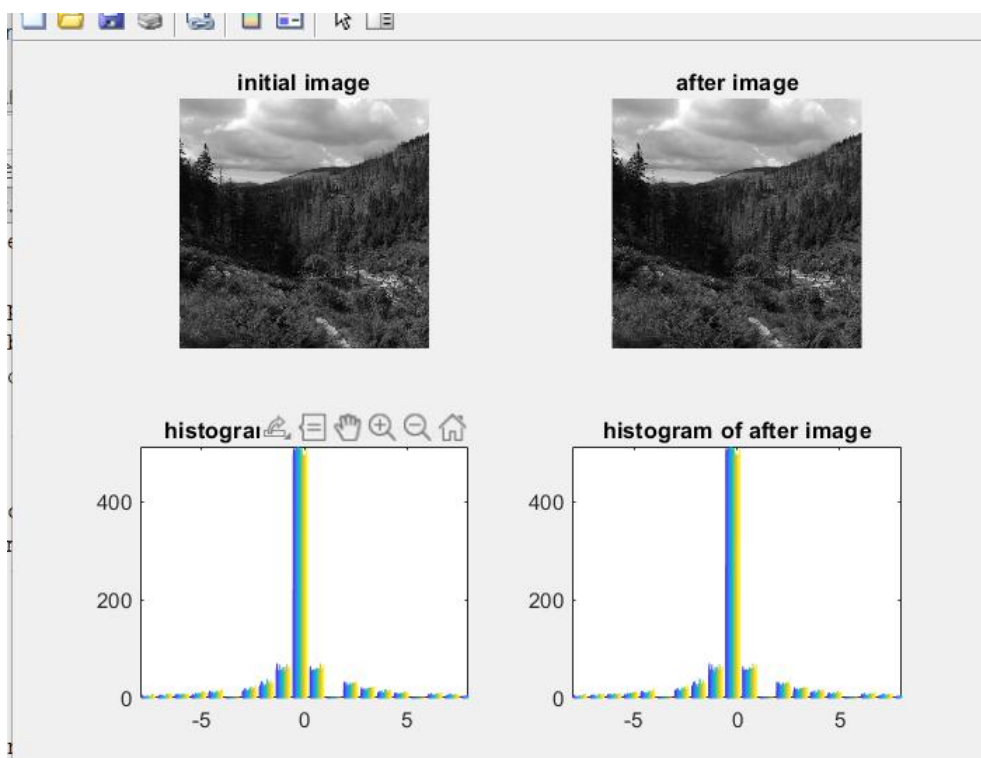


图 5.3.3

F5 算法基本保留了载体图像的直方图特性,所以能够抵抗视觉攻击和统计攻击。我在网上查阅资料发现, F5 算法的嵌入率可以达到 JPEG 图像的 13%。

### 5.3.2 嵌入信息提取

提取秘密信息的时候,先得到图像量化后的 DCT 系数,根据预先设定的密钥产生伪随机序列对 DCT 系数进行处理,然后根据设定的规则提取加密信息。

F5 的嵌入方式和 F4 相同，如图 5.3.4 所示

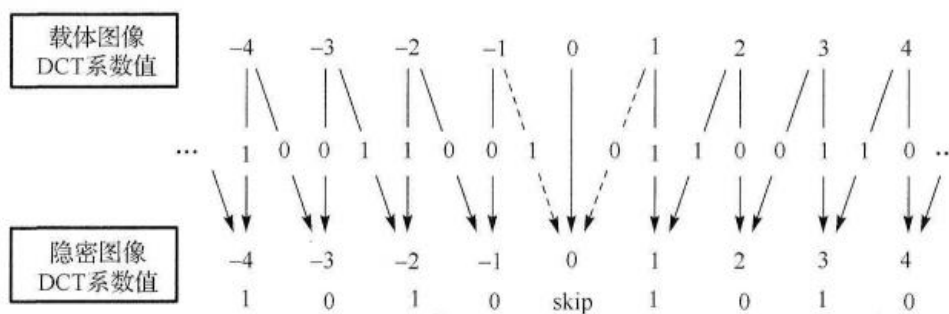


图 5.5.4

实验效果见下图 5.3.5 所示



图 5.3.5

方便起见，我只提取了前 800 比特的嵌入信息。

## 5.4 实验总结

首先, 在 F5 隐藏算法中不再使用 LSB 翻转嵌入方式, 嵌入秘密信息的原始图像 DCT 系数值的绝对值保持不变或者减 1, 这将保持全局直方图的形状不会由于嵌入秘密信息而改变; 其次, F5 隐藏算法不跳过值为 +1 或 -1 的 DCT 系数, 当某个秘密信息比特的嵌入使得 DCT+1, -1 变为 0 时, 则将该秘密信息比特在下一个 DCT 系数中重新嵌入, 因此接收方在提取秘密信息时只需要从非零系数中提取即可, 这种现象称为“收缩现象”。

本次实验采用的 nsf5 算法，所以收缩现象并不明显。

## 6 实验总结感想

第一个 LSB 实验是一个入门的实验，实验原理非常简单，就算没有听过课，单单看 PPT 都会做，后面的实验 2，主要的核心思想是建立在讲 JPEG 图像压缩那一块，其实实验的难度已经大大下降了，没有让我们自己动手去写如何提取 DCT 系数和量化，以及 zig-zag 扫描。老师以及给了封装好的工具。我自己个人对多媒体安全颇感兴趣，所以我每节课都有认真听讲，课上遇到不懂的问题，课下通过查阅资料，一一进行查缺补漏，所以实验做起来还是很顺利的，没有什么很大的困难。

通过真正的上手实验，我对于多媒体安全更加感兴趣，或者进一步说，这门课让我有了研究生学习多媒体安全的想法。

除此之外，我还感受到了数学对于这门学科的巨大作用，最经典也是最基础的 DCT 离散余弦变化，二维 DCT 变换就是将二维图像从空间域转换到频率域，通过之型扫描，一下就将二维可以转化成一维度。

最后感谢老师的辛苦付出。也谢谢这门课让我找到了感兴趣的方向。

## 参考文献

- [1] 孔祥维等.多媒体信息安全实践教程.科学出版社

指导教师评定意见

### 一、对实验报告的评语

--

## 二、对实验报告评分

评分项目 (分值)	程序内容 (36.8 分)	程序规范 (9.2 分)	报告内容 (36.8 分)	报告规范 (9.2 分)	考勤 (8 分)	逾期扣分	合 计 (100 分)
得分							