

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： 网安 1903 班

学 号： U201910471

姓 名： 张二勇

指导教师： 马晓静

报告日期： 2022 年 4 月 19 日

网络空间安全学院

目 录

1 JPEG 图像变换域信息隐藏实现	2
1 实验目的	2
2 实现思路	错误！未定义书签。
1.2 JSTEG 算法	2
1.2.1 算法描述	2
1.2.2 实现思路	2
1.2.3 实验结果及性能分析	3
1.3 F4 算法	4
1.3.1 算法描述	4
1.3.2 实现思路	5
1.3.3 实验结果及性能分析	5
1.4 F5 算法	5
1.4.1 算法描述	7
1.4.2 实现思路	7
1.4.3 实验结果及性能分析	8
3 实验小结	9
指导教师评定意见	10
一、对实验报告的评语	10
二、对实验报告评分	10
附录 A JPEG 图像变换域信息隐藏实现的源程序	11

1 JPEG 图像变换域信息隐藏实现

1 实验目的

通过实验达到：

- (1) 加深对变换域信息隐藏算法原理的理解；
- (2) 熟悉数字图片 JPEG 压缩格式；
- (3) 比较不同变换域信息隐藏算法在嵌入前后的 DCT 系数直方图特征。

2 JSTEG 算法

2.1 算法描述

JSteg 是最早以 JPEG 图像为载体的隐秘算法，主要是利用了 LSB 替换思想在 DCT 域的实现。

其主要思路是：将一个二进制位的隐秘信息嵌入到量化后的 DCT 系数的 LSB 上，但对原始值为-1，0，1 的 DCT 系数例外；

提取隐秘信息时，只需将载密图像中不等于-1，0，1 的量化 DCT 系数的 LSB 逐一取出即可；

顺序或随机选择元素；

(如图 2-1 所示)

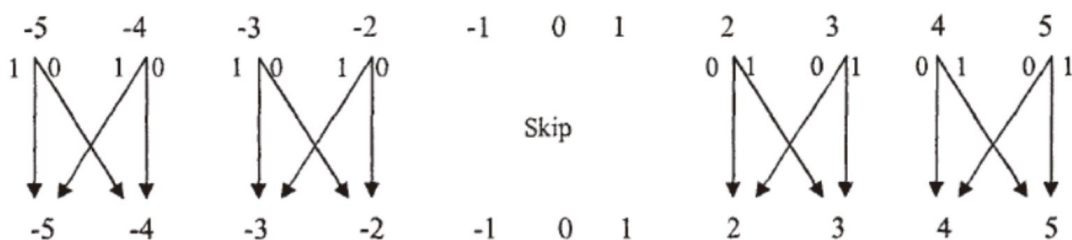


图 2-1 JSTEG 算法描述

2.2 实现思路

首先将图像以 JPEG 形式读入并将其做 DCT 域的变换，然后是 DCT 域系数和秘密信息比特流进行对应的匹配，值时 0，1，-1 的秘密信息比特流则自动跳过，其他的秘密信息比特则对进行取余，得到的余数如果和比特流的数字相同就不变，

负奇数遇到 1 就+1，负偶数遇到 0 就-1，正奇数遇到 0，就-1，正偶数遇到 1 就+1。

之后，再将加密了信息后的 DCT 系数转换回空域并且写入新的图片，再通过 histogram 函数统计 DCT 系数值，并且对比，从而得出值对现象；

提取信息，则 0，-1，1 跳过，其他则为正偶数和负奇数代表秘密信息比特 0，负偶数和正奇数代表秘密信息比特 1；

2.3 实验结果及性能分析

实验结果：

录入信息如下图 2-3-1 所示，将老师所给文件中的注意事项文件复制三遍：



图 2-3-1 录入信息

原图像和加密后图像及 DCT 系数直方图对比如图 2-3-2 所示，看不出变化（录入了两万多个 bit 的信息）：

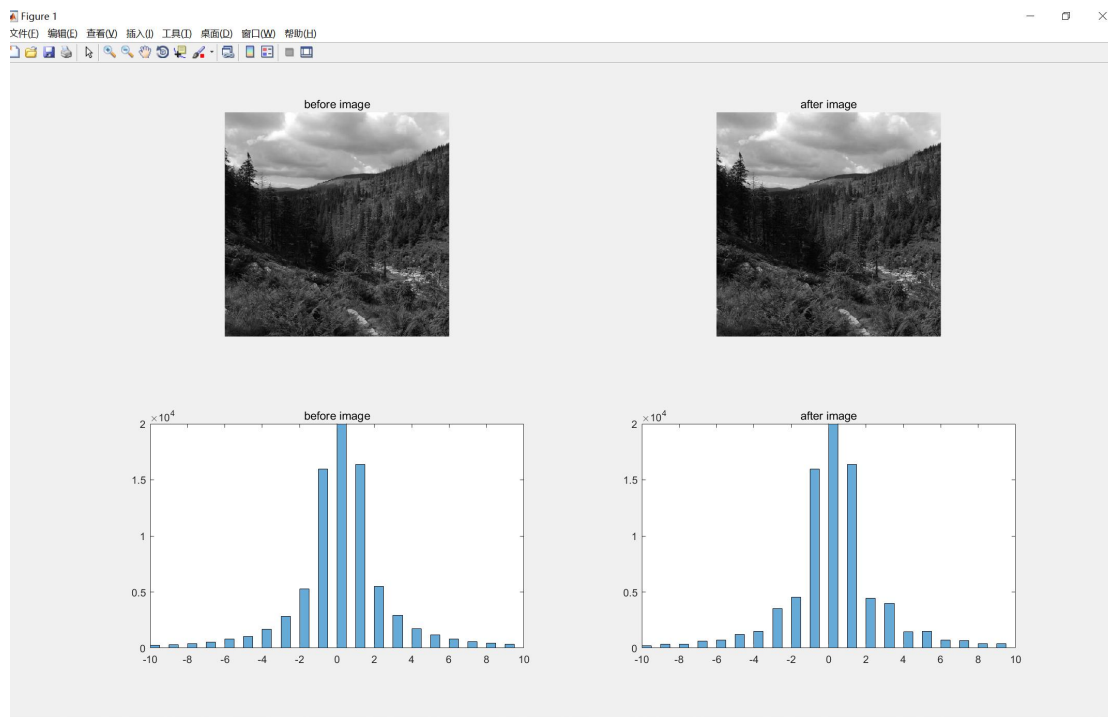


图 2-3-2 JPTeg 算法对比

如上图 2-3-2 所示，原图像和录入信息后图像基本无变化，DCT 系数直方图满足以下三个特性：

- (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；

且显然可以看出，除去-1, 1, 0 之外的 DCT 系数，在录入信息后出现明显的值对现象；

3 F4 算法

3.1 算法描述

嵌入：只修改不为 0 的 DCT 系数；

若不匹配则绝对值减 1，符号不变，避免“成对效应”；

若原 DCT 系数为+1，而待嵌入秘密比特为 0，则原系数会变为 0，本次嵌入操作无效，重新选择嵌入位；

若原 DCT 系数为 11，而待嵌入秘密比特为 1，则原系数会变为 0，本次嵌

入操作无效，重新选择嵌入位；

提取：将图像中不为 0 的 DCT 系数最低比特位按序取出；

其中：用正奇数和负偶数代表秘密消息 1，负奇数和正偶数代表秘密消息 0；

(如图 3-1 所示)

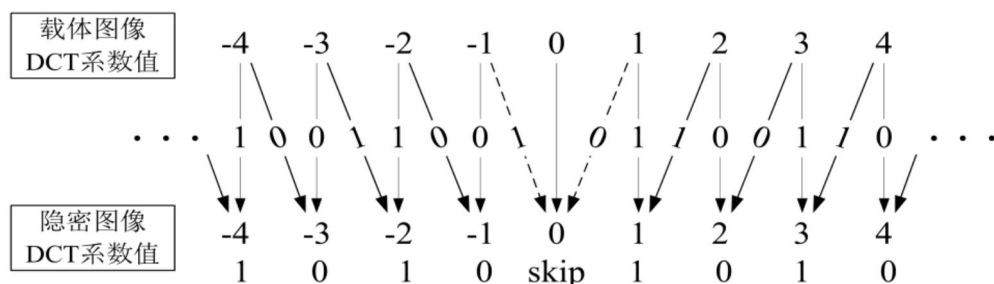


图 3-1-1 F4 算法描述

3.2 实现思路

同样将图像以 JPEG 形式读入并将其做 DCT 域的变换，然后是 DCT 域系数和秘密信息比特流进行对应的匹配，当 DCT 系数为 0 则自动跳过，为-1 时，录入比特为 1，则变成 0，为+1 时，录入比特位 0，则变成 0；为除 1 之外的正数，正偶数遇 1 则-1，遇 0 则不变；正奇数遇 0 则-1，遇 1 则不变；为除-1 之外的负数时，负奇数遇 1 则+1，遇 0 则不变；负偶数遇 0+1，遇 0 则不变；

之后，再将加密了信息后的 DCT 系数转换回空域并且写入新的图片，再通过 histogram 函数统计 DCT 系数值，并且对比。

提取信息时：遇到 0 则跳过，正偶数和负奇数代表秘密信息比特 0，正奇数和负偶数代表秘密信息比特 1；

3.3 实验结果及性能分析

实验结果：

录入信息如下图 2-3-1 所示，将老师所给文件中的注意事项文件复制三遍：



图 3-3-1 录入信息

原图像和加密后图像及DCT系数直方图对比如图 3-3-2 所示，看不出变化（录入了两万多个 bit 的信息）：

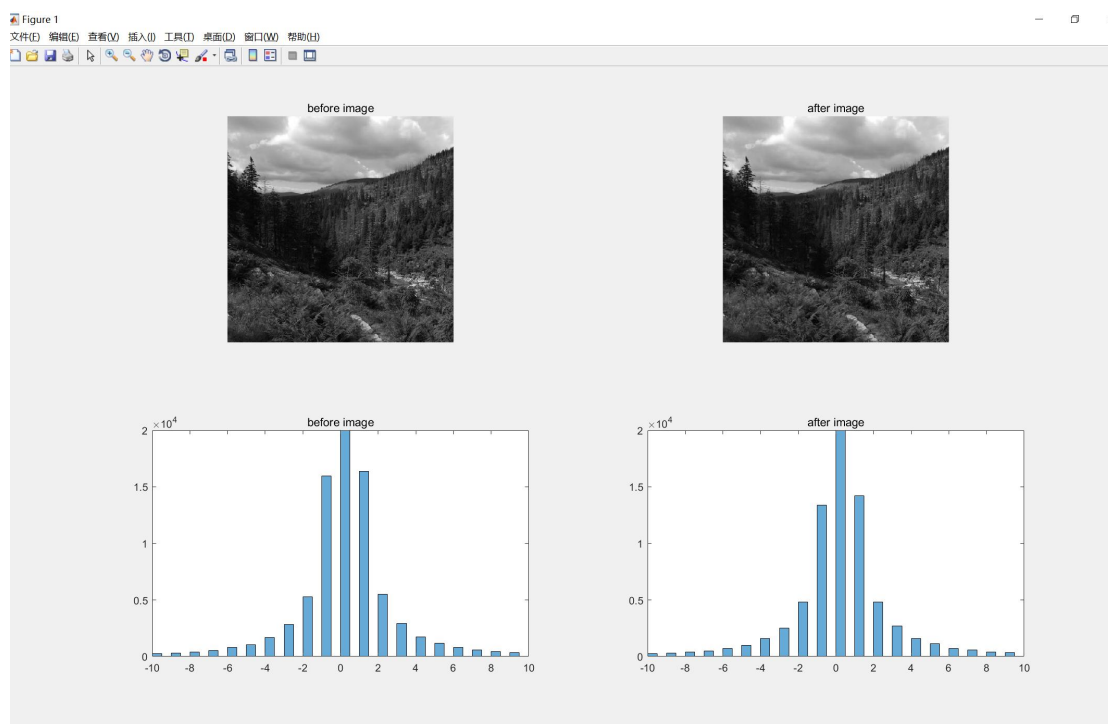


图 3-3-2 F4 算法对比

如上图 2-3-2 所示，原图像和录入信息后图像基本无变化，且与 JSTEG 相比，-1 和 1 的 DCT 系数的个数下降，因为 JSTEG 中-1, 1 则跳过，所以不变，而 F4

算法中个数会下降；

4 F5 算法

4.1 算法描述

F5 隐密算法是在 F4 的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 将其改成了随机嵌入信息，其次 F5 使用了矩阵编码的思想。

矩阵编码的思想：(1, n, k)，将 K 比特秘密信息嵌入到 $n=2^k-1$ 个符合要求的 DCT 系数上，最多只修改一个比特；

下面用样例简单说明矩阵编码的原理，且算法中使用的矩阵编码也是 3 嵌 2；

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 $b1=a1 \oplus a3$, $b2=a2 \oplus a3$ 。

修改数据为将 DCT 系数绝对值减一；

F5 算法优点：

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。

特殊情况：判断修改后的 DCT 系数是否变为 0，若没有则继续嵌入下一组数据；若系数变为 0，则本次隐藏无效，返回继续嵌入本组数据；

4.2 实现思路

首先将图像以 JPEG 形式读入并将其做 DCT 域的变换，然后设立数组将 DCT 系数流中所有不为 0 的 DCT 系数提取出来，并记录下它在矩阵中的位置和 DCT 系数，以及 DCT 系数代表的秘密信息（1 或者 0）；

之后，将其通过滑动窗口设立分组，滑动窗口大小为 3，并依据矩阵编码

分别判断是否需要对于当前分组中 3 个 DCT 系数进行修改；

特殊情况的考虑：对于分组中元素我们在进行所有修改操作前先对其分组中需要修改且修改后为 1 的 DCT 系数进行相应的判断，如果修改后的值为 0，则将该 DCT 系数变为 0，而当前分组中该 DCT 系数之后的 DCT 系数的窗口整体向后平移（有三种情况，分别为第一个，第二个，第三个 DCT 系数的变化），并跳过之后的修改操作，重新进入循环，知道特殊情况不出现为止，再对窗口中的 DCT 系数按照矩阵编码进行修改；

提取信息时，则同样设立数组将 DCT 系数流中所有不为 0 的 DCT 系数提取出来，并记录下它在矩阵中的位置和 DCT 系数，以及 DCT 系数代表的秘密信息（1 或者 0），并对其按三个一组进行分组，通过矩阵编码逆操作得出秘密信息；

4.3 实验结果及性能分析

实验结果：

录入信息如下图 4-3-1 所示，将老师所给文件中的注意事项文件复制三遍：



图 4-3-1 录入信息

原图像和加密后图像及 DCT 系数直方图对比如图 4-3-2 所示,看不出变化(录入了两万多个 bit 的信息)：

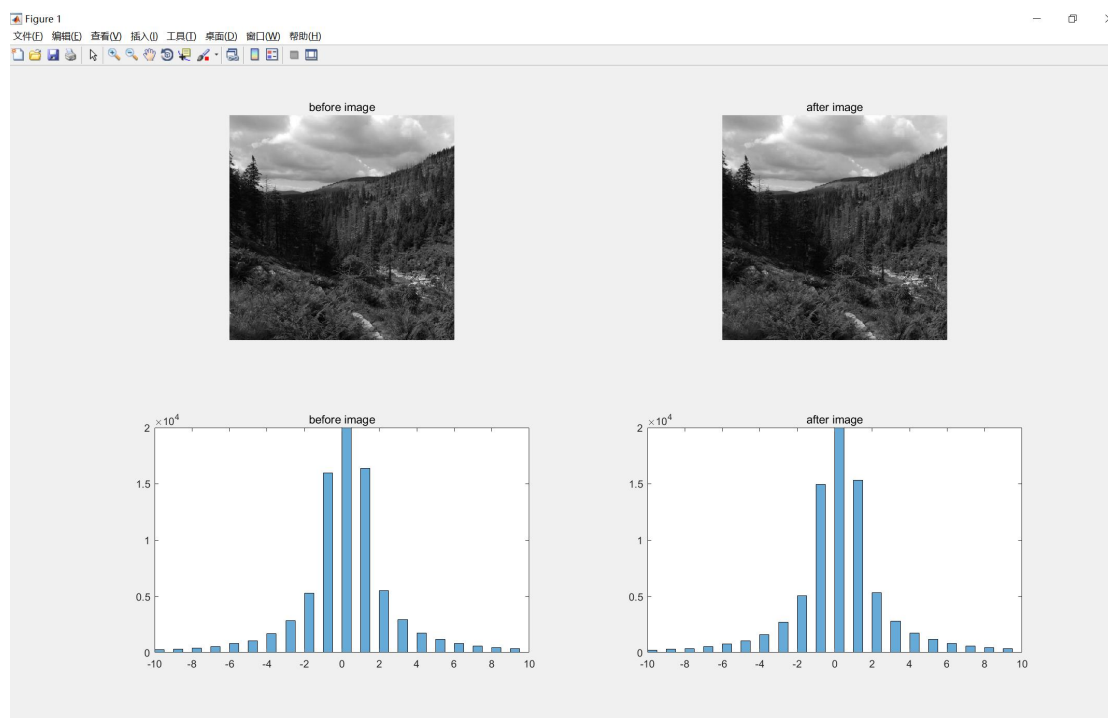


图 4-3-2 F5 算法对比

如上图 4-3-2 所示，原图像和录入信息后图像基本无变化，且 DCT 系数直方图原图像和载入秘密信息后的图像也无较大差别；所以能够抵抗视觉攻击和统计攻击；

3 实验小结

通过本次实验，我对一般的图像信息隐藏算法有了基本的了解，例如 LSB, JSTEG, F4, F5，并且通过编程对这几种算法实现简单的信息载入和信息提取的应用，且老师提供的代码有非常大的参考价值而且实现较为自由，尤其时 F5 的实现，让我对于 matlab 的编程有了一定的了解，希望以后能延长一下实验课的学时，能够对更多的算法有上手的应用；

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							

附录 A JPEG 图像变换域信息隐藏实现的源程序

LSB 加密程序

```
clear;
length=21904;
picture = imread('cover.jpg');
% image(picture);
double_picture = picture;
double_picture = double(double_picture);
zhang.txt_id = fopen('注意事项.txt','r');
% disp(isequal(zhang.txt_id,bitxor(zhang.txt_id,1)));
[msg2,len] = fread(zhang.txt_id,'ubit1');
one = ones(length,1);
msg = bitxor(msg2,one);
disp(isequal(msg,msg2));
[m,n] = size(double_picture);
p=1;
for f2 = 1:n
    for f1 = 1:m
        double_picture(f1,f2) =
double_picture(f1,f2)-mod(double_picture(f1,f2),2)+msg(p,1);
        if p == len
            break;
        end
        p=p+1;
    end
    if p==len
        break;
    end
end
double_picture=uint8(double_picture);
imwrite(double_picture,'stego.bmp');
subplot(121);imshow(picture);title('未嵌入信息的图像');
subplot(122);imshow(double_picture);title('嵌入信息的图像');
fclose(zhang.txt_id);

picture = picture(:);
picture = picture(1:21904,:);
double_picture = double_picture(:);
```

```
double_picture = double_picture(1:21904,:);

tab1 = tabulate(picture(:));
tab2 = tabulate(double_picture(:));
figure;
bar(tab1(150:170,1),tab1(150:170,2)),title('before image');
figure;
bar(tab2(150:170,1),tab2(150:170,2)),title('after image');
disp(sum(abs(picture-double_picture)));
```

LSB 解密程序

```
picture2 = imread('stego.bmp');
picture2 = double(picture2);
[m,n] = size(picture2);
frr = fopen('message.txt','a');
len = length;
p = 1;
for f2=1:n
    for f1=1:m
        if bitand(picture2(f1,f2),1) == 0
            fwrite(frr,1,'ubit1');
            result(p,1) = 1;
        else
            fwrite(frr,0,'ubit1');
            result(p,1) = 0;
        end
        if p == len
            break;
        end
        if p < len
            p=p+1;
        end
    end
    if p == len
        break;
    end
end
fclose(frr);
disp(isequal(double_picture,picture2));
disp(isequal(result,msg2));
```

JSTEG 加密程序

```

cover='cover.jpg';
stego='jsteg.jpg';
wen.txt_id=fopen('注意事项.txt','r');
[msg,L]=fread(wen.txt_id,'ubit1');
try
    jpeg_cover=jpeg_read(cover);
    dct_cover=jpeg_cover.coef_arrays{1};
    dct_cover2=jpeg_cover.coef_arrays{1};
catch
    error('Error');
end

len=length(msg);
sum=1;
[m,n]=size(dct_cover);
for f2 =1:n
    for f1 =1:m
        if(abs(dct_cover(f1,f2))<=1)
            continue;
        end
        if(dct_cover(f1,f2)>1)
            odd=mod(dct_cover(f1,f2),2);
            if(msg(sum,1)==0&&odd==1)
                dct_cover(f1,f2)=dct_cover(f1,f2)-1;
            end
            if(msg(sum,1)==1 && odd==0)
                dct_cover(f1,f2)=dct_cover(f1,f2)+1;
            end
        end
        if(dct_cover(f1,f2)<-1)
            odd=abs(mod(dct_cover(f1,f2),2));
            if(msg(sum,1)==1&&odd==1)
                dct_cover(f1,f2)=dct_cover(f1,f2)+1;
            end
            if(msg(sum,1)==0 && odd==0)
                dct_cover(f1,f2)=dct_cover(f1,f2)-1;
            end
        end
        if(sum==len)
            break;
        end
        sum=sum+1;
    end
end

```

```

        end
        if sum == len
            break;
        end
    end
end
try
    jpeg_cover.coef_arrays{1}=dct_cover;
    jpeg_cover.optimize_coding=1;
    jpeg_write(jpeg_cover,stego);
catch
    error('ERROR2')
end
subplot(2,2,1);
imshow(cover);
title('before image');
subplot(2,2,2);
imshow(stego);
title('after image');

subplot(2,2,3);
histogram(dct_cover2);
axis([-10 10,0 2*1e4]);
title('before image');
subplot(2,2,4);
histogram(dct_cover);
axis([-10 10,0 2*1e4]);
title('after image');
```

JSTEG 解密程序

```

cover='cover.jpg';
stego='jsteg.jpg';
fr=fopen('meg_jsteg.txt','a');
try
    jpeg_stego=jpeg_read(stego);
    dct_stegot=jpeg_stego.coef_arrays{1};
catch
    error('Error');
end
len=21904;
sum=1;
[m,n]=size(dct_stegot);
```

```

for f2 =1:n
    for f1 =1:m
        if(abs(dct_stegot(f1,f2))<=1)
            continue;
        end
        if(dct_stegot(f1,f2)>1)
            odd=mod(dct_stegot(f1,f2),2);
            if(odd==0)
                fwrite(frr,0,'ubit1');
            end
            if( odd==1)
                fwrite(frr,1,'ubit1');
            end
        end
        if(dct_stegot(f1,f2)<-1)
            odd=abs(mod(dct_stegot(f1,f2),2));
            if(odd==0)
                fwrite(frr,1,'ubit1');
            end
            if(odd==1)
                fwrite(frr,0,'ubit1');
            end
        end
        if(sum==len)
            break;
        end
        sum=sum+1;
    end
end
if sum ==len
    break;
end
end
fclose(frr);

```

F4 加密程序

```

cover='cover.jpg';
stego='F4.jpg';
wen.txt_id=fopen('注意事项.txt','r');
[msg,L]=fread(wen.txt_id,'ubit1');
try
    jpeg_cover=jpeg_read(cover);
    dct_cover=jpeg_cover.coef_arrays{1};

```



```

    dct_cover2=jpeg_cover.coef_arrays{1};
catch
    error('Error');
end

len=length(msg);
sum=1;
[m,n]=size(dct_cover);
for f2 =1:n
    for f1 =1:m
        if((dct_cover(f1,f2))==0)
            continue;
        end
        if(dct_cover(f1,f2)==1 &&msg(sum,1)==0)
            dct_cover(f1,f2)=0;
            continue;
        end

        if(dct_cover(f1,f2)>=1)
            odd=mod(dct_cover(f1,f2),2);
            if(msg(sum,1)~=odd)
                dct_cover(f1,f2)=dct_cover(f1,f2)-1;
            end
        end
        if(dct_cover(f1,f2)==-1 &&msg(sum,1)==1)
            dct_cover(f1,f2)=0;
            continue;
        end
        if(dct_cover(f1,f2)<=-1)
            odd=abs(mod(dct_cover(f1,f2),2));
            if(msg(sum,1)==odd)
                dct_cover(f1,f2)=dct_cover(f1,f2)+1;
            end
        end
        end
        if(sum==len)
            break;
        end
        sum=sum+1;
    end
    if sum ==len
        break;
    end
end
end

```

```

try
    jpeg_cover.coef_arrays{1}=dct_cover;
    jpeg_cover.optimize_coding=1;
    jpeg_write(jpeg_cover,stego);
catch
    error('ERROR2')
end
subplot(2,2,1);
imshow(cover);
title('before image');
subplot(2,2,2);
imshow(stego);
title('after image');

subplot(2,2,3);
histogram(dct_cover2);
axis([-10 10,0 2*1e4]);
title('before image');
subplot(2,2,4);
histogram(dct_cover);
axis([-10 10,0 2*1e4]);
title('after image');

```

F4 解密程序

```

cover='cover.jpg';
stego='F4.jpg';
fr=fopen('msg_F41.txt','a');
try
    jpeg_stego=jpeg_read(stego);
    dct_stego=jpeg_stego.coef_arrays{1};
catch
    error('Error');
end
len=21904;
sum=1;
[m,n]=size(dct_stego);
for f2 =1:n
    for f1 =1:m
        if(dct_stego(f1,f2)==0)
            continue;
        end
    end
end

```

```

        if dct_stego(f1,f2)>0
            odd=mod(dct_stego(f1,f2),2);
            fwrite(frr,odd,'ubit1');
        else
            odd=mod(dct_stego(f1,f2),2);
            fwrite(frr,mod(odd+1,2),'ubit1');
        end
        if(sum==len)
            break;
        end
        sum=sum+1;

    end
    if sum ==len
        break;
    end
end
fclose(frr);

```

F5 加密程序

```

cover='cover.jpg';
stego='F5.jpg';
wen.txt_id=fopen('1.txt','r');
[msg,L]=fread(wen.txt_id,'ubit1');
try
    jpeg_cover=jpeg_read(cover);
    dct_cover=jpeg_cover.coef_arrays{1};
    dct_cover2=jpeg_cover.coef_arrays{1};
catch
    error('Error');
end

len=length(msg);
sum=1;
[m,n]=size(dct_cover);
id = 1;
x=zeros(1,512*512);
y=zeros(1,512*512);
num_cover=zeros(1,512*512);
bit_cover=zeros(1,512*512);
condition_cover=zeros(1,512*512);
len2=512*512;

```

```

for f2 =1:n
    for f1 =1:m
        if((dct_cover(f1,f2))==0)
            continue;
        end
        if((dct_cover(f1,f2))>0)
            x(1,id)=f1;
            y(1,id)=f2;
            num_cover(1,id)=dct_cover(f1,f2);
            odd=mod(dct_cover(f1,f2),2);
            condition_cover(1,id)=odd;
            id=id+1;
            disp(dct_cover(f1,f2));
        end
        if((dct_cover(f1,f2))<0)
            x(1,id)=f1;
            y(1,id)=f2;
            num_cover(1,id)=dct_cover(f1,f2);
            odd=mod(dct_cover(f1,f2),2);
            condition_cover(1,id)=bitxor(odd,1);
            disp(dct_cover(f1,f2));
            id=id+1;

        end
        if(sum==len2)
            break;
        end
        sum=sum+1;
    end
    if sum ==len2
        break;
    end
end
i=1
flag=0;
now= zeros(1,3);
now2 = zeros(1,3);
sum2=1;
now(1,1)=1;
now(1,2)=2;
now(1,3)=3;
while i<=id&&sum2<=len
    disp(dct_cover(x(1,now(1,1)),y(1,now(1,1))));
    disp(dct_cover(x(1,now(1,2)),y(1,now(1,2))));

```

```

disp(dct_cover(x(1,now(1,3)),y(1,now(1,3))));
disp(msg(sum2,1));
disp(msg(sum2+1,1));
if abs(dct_cover(x(1,now(1,1)),y(1,now(1,1))))==1
    if
msg(sum2,1)~=bitxor(condition_cover(1,now(1,1)),condition_cover(1,now(1,2)))&&
msg(sum2+1,1)==bitxor(condition_cover(1,now(1,2)),condition_cover(1,now(1,3)))
        dct_cover(x(1,now(1,1)),y(1,now(1,1)))=0;
        now(1,1)=now(1,2);
        now(1,2)=now(1,3);
        now(1,3)=now(1,3)+1;
        flag=flag+1;
        continue;
    end
end
if abs(dct_cover(x(1,now(1,2)),y(1,now(1,2))))==1
    if
msg(sum2,1)~=bitxor(condition_cover(1,now(1,1)),condition_cover(1,now(1,2)))&&
msg(sum2+1,1)~=bitxor(condition_cover(1,now(1,2)),condition_cover(1,now(1,3)))
        dct_cover(x(1,now(1,2)),y(1,now(1,2)))=0;
        now(1,2)=now(1,3);
        now(1,3)=now(1,3)+1;
        flag=flag+1;
        continue;
    end
end
if abs(dct_cover(x(1,now(1,3)),y(1,now(1,3))))==1
    if
msg(sum2,1)==bitxor(condition_cover(1,now(1,1)),condition_cover(1,now(1,2)))&&
msg(sum2+1,1)~=bitxor(condition_cover(1,now(1,2)),condition_cover(1,now(1,3)))
        dct_cover(x(1,now(1,3)),y(1,now(1,3)))=0;
        now(1,3)=now(1,3)+1;
        flag=flag+1;
        continue;
    end
end
if
msg(sum2,1)~=bitxor(condition_cover(1,now(1,1)),condition_cover(1,now(1,2)))&&
msg(sum2+1,1)==bitxor(condition_cover(1,now(1,2)),condition_cover(1,now(1,3)))
    if (dct_cover(x(1,now(1,1)),y(1,now(1,1))))>0

dct_cover(x(1,now(1,1)),y(1,now(1,1)))=dct_cover(x(1,now(1,1)),y(1,now(1,1)))-1;
end
    if (dct_cover(x(1,now(1,1)),y(1,now(1,1))))<0

```

```

dct_cover(x(1,now(1,1)),y(1,now(1,1)))=dct_cover(x(1,now(1,1)),y(1,now(1,1)))+1;
    end
    end
    if
msg(sum2,1)~=bitxor(condition_cover(1,now(1,1)),condition_cover(1,now(1,2)))&&
msg(sum2+1,1)~=bitxor(condition_cover(1,now(1,2)),condition_cover(1,now(1,3)))
        if (dct_cover(x(1,now(1,2)),y(1,now(1,2))))>0

dct_cover(x(1,now(1,2)),y(1,now(1,2)))=dct_cover(x(1,now(1,2)),y(1,now(1,2)))-1;
        end
        if (dct_cover(x(1,now(1,2)),y(1,now(1,2))))<0

dct_cover(x(1,now(1,2)),y(1,now(1,2)))=dct_cover(x(1,now(1,2)),y(1,now(1,2)))+1;
        end
        end
        if
msg(sum2,1)==bitxor(condition_cover(1,now(1,1)),condition_cover(1,now(1,2)))&&
msg(sum2+1,1)~=bitxor(condition_cover(1,now(1,2)),condition_cover(1,now(1,3)))
            if (dct_cover(x(1,now(1,3)),y(1,now(1,3))))>0

dct_cover(x(1,now(1,3)),y(1,now(1,3)))=dct_cover(x(1,now(1,3)),y(1,now(1,3)))-1;
            end
            if (dct_cover(x(1,now(1,3)),y(1,now(1,3))))<0

dct_cover(x(1,now(1,3)),y(1,now(1,3)))=dct_cover(x(1,now(1,3)),y(1,now(1,3)))+1;
            end
            end
            sum2=sum2+2;
            disp(dct_cover(x(1,now(1,1)),y(1,now(1,1))));
            disp(dct_cover(x(1,now(1,2)),y(1,now(1,2))));
            disp(dct_cover(x(1,now(1,3)),y(1,now(1,3))));
            now(1,1)=now(1,3)+1;
            now(1,2)=now(1,3)+2;
            now(1,3)=now(1,3)+3;
            i=now(1,1);
        end
    end

try
    jpeg_cover.coef_arrays{1}=dct_cover;
    jpeg_cover.optimize_coding=1;

```

```

        jpeg_write(jpeg_cover,stego);
    catch
        error('ERROR2')
    end
    subplot(2,2,1);
    imshow(cover);
    title('before image');
    subplot(2,2,2);
    imshow(stego);
    title('after image');

    subplot(2,2,3);
    histogram(dct_cover2);
    axis([-10 10,0 2*1e4]);
    title('before image');
    subplot(2,2,4);
    histogram(dct_cover);
    axis([-10 10,0 2*1e4]);
    title('after image');
    %disp(isequal(dct_cover,dct_cover2));

```

F5 解密程序

```

stego='F5.jpg';

len=10304;
frr=fopen('msg_F5.txt','a');

try
    jpeg_stego = jpeg_read(stego);
    dct_stego = jpeg_stego.coef_arrays{1};
catch
    error('ERROR');
end
[m,n]=size(dct_stego);
idD=1;
id=0;
id = 1;
x=zeros(1,512*512);
y=zeros(1,512*512);
num_stego=zeros(1,512*512);
bit_stego=zeros(1,512*512);
condition_stego=zeros(1,512*512);

```

```

len2=512*512;
sum=1;
for f2 =1:n
    for f1 =1:m
        if((dct_stego(f1,f2))==0)
            continue;
        end
        if((dct_stego(f1,f2))>0)
            x(1,id)=f1;
            y(1,id)=f2;
            num_stego(1,id)=dct_stego(f1,f2);
            odd=mod(dct_stego(f1,f2),2);
            condition_stego(1,id)=odd;
            id=id+1;
        end
        if((dct_stego(f1,f2))<0)
            x(1,id)=f1;
            y(1,id)=f2;
            num_stego(1,id)=dct_stego(f1,f2);
            odd=mod(dct_stego(f1,f2),2);
            condition_stego(1,id)=bitxor(odd,1);
            id=id+1;
        end
        if(sum==len2)
            break;
        end
        sum=sum+1;
    end
    if sum ==len2
        break;
    end
end
disp(num_stego(1,1:40));
sum2=1;
msgnum=1;
message=zeros(1,len);
while msgnum<=len

message(1,msgnum)=bitxor(condition_stego(1,sum2),condition_stego(1,sum2+1));

message(1,msgnum+1)=bitxor(condition_stego(1,sum2+1),condition_stego(1,sum2+2)
);
    msgnum=msgnum+2;
    sum2=sum2+3;

```



```
end
disp(isequal(msg.',message));
for i=1:len
    fwrite(frr,message(1,i),'ubit1');
end
fclose(frr);
```

注：不同的算法使用的变量名有些相同，使用不同的算法前确保先执行 `clear` 命令；