**《数字内容安全》**

**实验报告**

**实验名称：** **时空域的信息隐藏和提取**   **任课教师：付宇**

**学号：**200340164  **姓名：**张博文  **提交日期：** 2022/11/2

**一、实验目标：**

实现 LSB 信息隐藏和提取

**二、实验设计：**

1、基本 LSB 信息隐藏及提取

（1）基本 LSB 信息隐藏

1）读入灰度图像或者读入彩色图像，并将彩色图像转化为灰度图像。

2）读入实验一中生成的加密后的二值水印图像。

3）更改二值水印图像大小，要求和载体图像大小一致。

4）将二值水印图像替换载体图像第 1 位平面（最低位；最不重要位），

利用 bitset 函数。

（2）基本 LSB 信息提取

1）利用 bitget 函数，提含有秘密图像的载体图像最低位，得到加密后的

二值水印图像。

2）对二值图像解密后得到原始水印图像。

2、随机选择嵌入位进行 LSB 信息隐藏及提取

2

信息隐藏和提取过程基本同上，水印图像按照原始大小嵌入，嵌入位置根据

随机间隔法选择，其伪代码如下:

随机间隔法的思想比较简单，主要是利用随机数的大小控制前后两个嵌入位

的距离。如一个长为 N 的服从 U(0,1)的随机序列 R={r1,r2,…,rN}，N 大于秘密信息

长度。取第一个嵌入位为 i，伪 C 代码描述有：

imbedding address=I;

for (j=1; j<=length(message); j++)

{

if (rj> 0.5 )

imbedding address + = k;

else

imbedding address + = p;

}

以上代码的实质就是通过判断相应的随机数与 0.5 的大小，若大于 0.5，则选

择的嵌入位与前一个嵌入位间隔 k-1 位，否则间隔 p-1 位。

3、分别计算含有秘密信息的载体图像的峰值信噪比（PSNR），分析信息隐藏

图像质量。

PSNR 计算公式如下：

4、分别对含有秘密信息的载体图像进行攻击（缩放 resize、jpeg 压缩、低通

滤波 filter2、中值滤波 medfilt2 等、剪切），计算其误码率。

误码率=提取后的秘密信息与原始信息相比的错误像素点数/图像总的像素

点数

**三、实验记录：**

1、基本 LSB 信息隐藏及提取

（1）基本 LSB 信息隐藏

1）读入灰度图像或者读入彩色图像，并将彩色图像转化为灰度图像。

2）读入实验一中生成的加密后的二值水印图像。

3）更改二值水印图像大小，要求和载体图像大小一致。

4）将二值水印图像替换载体图像第 1 位平面（最低位；最不重要位），

利用 bitset 函数。

代码如下：

%% 基本LSB信息隐藏

% 读入灰度图像挥着读入彩色图像，并将彩色图像转换成灰度图像

so=imread("1.jpg")

gray=rgb2gray(so);

subplot(3,3,1);

imshow(so);

title('彩色');

subplot(3,3,2);

imshow(gray);

title('灰度');

% 读入实验一生成的加密后的二值水印图像

e=imread('ezjm.jpg');

subplot(3,3,3);

imshow(e);

title('实验一');

% 更改二值水印图像的大小，要求与载体图像大小一致

% 将二值水印图像替换载体图像第 1 位平面（最低位；最不重要位），

% 利用 bitset 函数。

b1=bitset(gray,1,e);

subplot(3,3,4);

imshow(b1);

title('隐藏的图片');

（2）基本 LSB 信息提取

1）利用 bitget 函数，提含有秘密图像的载体图像最低位，得到加密后的

二值水印图像。

2）对二值图像解密后得到原始水印图像。

2、随机选择嵌入位进行 LSB 信息隐藏及提取

2

信息隐藏和提取过程基本同上，水印图像按照原始大小嵌入，嵌入位置根据

随机间隔法选择，其伪代码如下:

随机间隔法的思想比较简单，主要是利用随机数的大小控制前后两个嵌入位

的距离。如一个长为 N 的服从 U(0,1)的随机序列 R={r1,r2,…,rN}，N 大于秘密信息

长度。取第一个嵌入位为 i，伪 C 代码描述有：

imbedding address=I;

for (j=1; j<=length(message); j++)

{

if (rj> 0.5 )

imbedding address + = k;

else

imbedding address + = p;

}

以上代码的实质就是通过判断相应的随机数与 0.5 的大小，若大于 0.5，则选

择的嵌入位与前一个嵌入位间隔 k-1 位，否则间隔 p-1 位。

**代码如下：**

randlsbhide.m

% 函数功能: 本函数将完成随机选择LSB 的信息隐秘, 载体选用灰度BMP 图

function [ste\_cover, len\_total] = randlsbhide(input, file, output, key)

% 读入图像矩阵

cover = imread(input) ;

ste\_cover = cover;

ste\_cover = double(ste\_cover ) ;

% 将文本文件转换为二进制序列

f\_id = fopen(file, 'r') ;

[ msg, len\_total] = fread(f\_id, 'ubit1') ;

% 判断嵌入消息量是否过大

[ m, n] = size(ste\_cover) ;

if len\_total > m\*n

error('嵌入消息量过大, 请更换图像') ;

end

% p 作为消息嵌入位数计数器

p = 1;

% 调用随机间隔函数选取像素点

[ row, col] = randinterval(ste\_cover, len\_total, key) ;

% 在LSB 隐秘消息

for i = 1:len\_total

ste\_cover(row( i), col( i)) = ste\_cover(row( i) ,col(i) )-mod(ste\_cover(row( i) , col(i)),2 ) + msg(p, 1) ;

if p == len\_total

break;

end

p = p + 1;

end

ste\_cover = uint8(ste\_cover) ;

imwrite(ste\_cover, output) ;

% 显示实验结果

subplot(1, 2, 1) ; imshow(cover) ; title('原始图像') ;

subplot(1, 2, 2) ; imshow(output) ; title('隐藏信息的图像') ;

信息提取

randlsbget.m

% 函数功能: 本函数将完成提取隐秘于LSB 上的秘密消息

function result = randlsbget( output, len\_total, goalfile, key)

ste\_cover = imread( output) ;

ste\_cover = double( ste\_cover ) ;

% 判断嵌入消息量是否过大

[ m, n] = size( ste\_cover ) ;

frr = fopen( goalfile, 'a') ;

% p 作为消息嵌入位数计数器, 将消息序列写回文本文件

p = 1;

% 调用随机间隔函数选取像素点

[ row, col] = randinterval( ste\_cover , len\_total, key) ;

for i = 1:len\_total

if bitand( ste\_cover( row( i) , col( i) ) , 1) == 1

fwrite( frr, 1, 'ubit1') ;

result( p, 1 ) = 1;

else

fwrite( frr, 0, 'ubit1') ;

result( p, 1 ) = 0;

end

if p == len\_total

break;

end

p = p + 1;

end

fclose( frr) ;

自动脚本

%% 隐藏

[ste\_cover,len\_total] = randlsbhide('lena.bmp','1.txt','output.bmp',2022)

%% 提取

result = randlsbget( 'output.bmp', len\_total, '2.txt', 2022);

3、分别计算含有秘密信息的载体图像的峰值信噪比（PSNR），分析信息隐藏

图像质量。

代码如下：

%分别计算含有秘密信息的载体图像的峰值信噪比（PSNR），分析信息隐藏

%图像质量。

j=imread("2-2.jpg");

g=imread('ezjm.jpg');

% 直接调用psnr api

peaksnr=psnr(g,j);

disp(peaksnr);

此处可以直接调用计算峰值信噪比的api

%分别计算含有秘密信息的载体图像的峰值信噪比（PSNR），分析信息隐藏

%图像质量。

j=imread("output.bmp");

g=imread('lena.bmp');

% 直接调用psnr api

peaksnr=psnr(g,j);

disp(peaksnr);



4、分别对含有秘密信息的载体图像进行攻击（缩放 resize、jpeg 压缩、低通

滤波 filter2、中值滤波 medfilt2 等、剪切），计算其误码率。

误码率=提取后的秘密信息与原始信息相比的错误像素点数/图像总的像素

点数

%分别对含有秘密信息的载体图像进行攻击（缩放 resize、jpeg 压缩、低通滤波 filter2、中值滤波 medfilt2 等、剪切），计算其误码率。

% 4.误码率

% 攻击，选取上述实验生成的秘密信息载体bitset0

% 原始秘密信息为上述实验生成的

% bitset0=imread("output.bmp");

biset0=b2;

bitset1=imresize(bitset0,[32,32]);

bitset1=imresize(bitset1,[64,64]);

bitset1=bitget(bitset1,1);

zzz=imread('ezjm.jpg');

% 计算误码率

count = 0;

for i=1:64\*64

if bitset1(i) == zzz(i)

count = count + 1;

end

end

WUMA = count/(64\*64);

disp(WUMA);



**四、实验思考与体会**

了解了LSB信息隐藏算法的原理。通过图片的权重分配，使得权重较小的位置可以藏匿我们想要的信息

对于低位来说：嵌入1个位平面时，受影响的值只有01；嵌入2个位平面时，受影响的值有04；嵌入3个位平面时，受影响的值也仅有011。这相对于常规图片的0255来说影响不大。

那么我们将嵌入照片的高位嵌入载体照片的地位，便可以保存嵌入照片的众多信息的同时对载体照片影响微乎其微。由此可以实现信息隐藏。