实验二 熟悉二值量化索引调制(QIM)水印过程

学号：201711123010 班级：信息安全2017级 姓名：谌雯馨

1. 实验目的

熟悉Matlab/c/python图像处理编程环境。

模拟并熟悉图像二值量化索引调制水印过程。

1. 实验环境

本次实验所使用的图像是 512x512 的灰度图像（即 lena512.bmp），使用的工具及版本为 MATLAB R2017B。

1. 实验内容

3.1 模拟数字像QIM水印嵌入过程

1. 实验过程

1.将待嵌入的水印信息转为二进制码流

2.读入原始图像载体，并转为一维数组，以便嵌入水印

3.通过matlab函数plot()将其图像画出

1. 定义量化器Q(-1)和Q(+1)
2. 进行水印嵌入过程：如果当前待嵌水印信息为0时，则使用Q(-1)进行量化；如果当前待嵌水印信息为1时，则使用Q(+1)进行量化。
3. 重新生成图片，该图片为掩密载体
4. 实验原理

基本原理是按照待嵌入的二进制水印比特b∈{+1, -1}，选择使用两个均匀量化器中的一

个对载体信号x进行量化处理 进行量化处理。

1. 实验代码（含详细注释）

close all;

clear all;

clc;

x=input('please input a string you want to Steganograph:','s');%输入字符串

lenth=length(x);%根据字符串长度嵌入

img=imread('lena.bmp');

[I\_x, I\_y] = size(img);

n=I\_x\*I\_y;%图像大小

imgg=img';

img\_one=double(imgg(:)');%将数组变为一维数组便于嵌字符串

img\_cox=[];

strin\_bits='';

strin\_bits=[lenth,x];%将字符串长度和字符串本身连接在一起

strin\_bits=str\_to\_bits(strin\_bits);%将字符串转为ASCII码的二进制值

w=[];

num=lenth\*8+8;%长度

for i=1:num

if strin\_bits(i)==0

w(i)=-1;

else

w(i)=1;

end

end

% 量化器

% Q1从0开始，Q2从d开始，步长为delta

d = 5;

delta = 3;

Q1 = 0:2\*delta:n; % -1

Q2 = d:2\*delta:n; % 1

% 嵌入水印

for i=1:length(w)

if w(i)==-1 %用Q(-1)进行量化

img\_one(i) = round((img\_one(i)-d)/2/delta)\*2\*delta+d;

else

img\_one(i) = round((img\_one(i)-d-delta)/2/delta)\*2\*delta+d+delta;%用Q(+1)进行量化

end

end

Y = reshape(img\_one, [I\_x, I\_y]);

Y=uint8(Y');

imwrite(Y,'lena\_watermark50.bmp');%生成新的图片

save\_figure = figure();

subplot(1,2,1),imshow(img),title('原始图像'),subplot(1,2,2),imshow(Y),title('嵌入水印后');

saveas(save\_figure, 'compare.bmp');

%将字符串转为二进制串函数

function [msg\_bits] = str\_to\_bits(msgStr);

msgBin = de2bi(int8(msgStr),8,'left-msb');

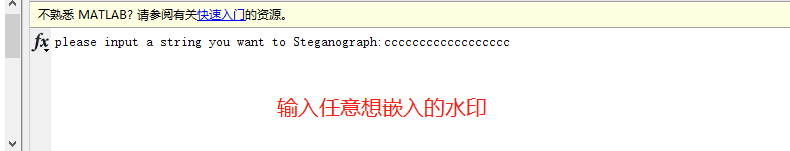
len = size(msgBin,1).\*size(msgBin,2);

msg\_bits = reshape(double(msgBin).',len,1).';

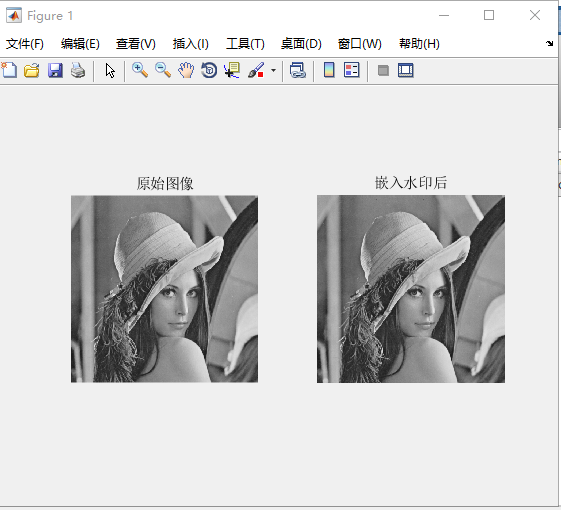
end

1. 实验结果
2. 输入一串任意想嵌入的字符串，长度不超过（512\*512/8-8），输出为一张嵌入水印的图片，从肉眼几乎无法判断是否嵌入水印

输入：



输出：



3.2 模拟数字像QIM水印提取过程

1. 实验过程
2. 读入含密载体图像，获得一个512\*512的矩阵
3. 获取量化器
4. 先提取前八位二进制码流，即获得嵌入字符串的长度，提取方式为通过计算Q(-1)和Q（+1）两种情况的欧式距离，并进行比较，距离短的即说明使用了该种方式量化，则码流为该数字
5. 根据获取的长度，将字符串提取，提取方式同上
6. 计算PSNR值，查看图像质量
7. 实验代码（含详细注释）

%% 提取水印

close all;

clear all;

clc;

img\_yuan=imread('lena.bmp');

I\_double=double(img\_yuan);

img=imread('lena\_watermark.bmp');%读取嵌入水印的图像

[I\_x, I\_y] = size(img);

n=I\_x\*I\_y;

imgg=img';

img\_one=double(imgg(:));%变为一维数组

%频域

%rY = dct2(Y).\*255;

%rY = rY(:);

% % 空域处理

% rY = double(Y(:));

% 量化器

% Q1从0开始，Q2从d开始，步长为delta

d = 5;

delta = 3;

Q1 = 0:2\*delta:n; % -1

Q2 = d:2\*delta:n; % 1

% 提取水印

leng = '';%提取水印长度

for i=1:8

b1 = norm(img\_one(i)-(round((img\_one(i)-d)/2/delta)\*2\*delta+d)); % pdist默认欧氏距离 euclidean

b2 = norm(img\_one(i)-(round((img\_one(i)-d-delta)/2/delta)\*2\*delta+d+delta));

if b1 < b2

leng=[leng,'0'];

else

leng=[leng,'1'];

end

end

num=bin2dec(leng);

cha\_num=num\*8+8;%根据长度提取信息

watermark='';

temp='';

for i=9:cha\_num%真正开始提取水印

b1 = norm(img\_one(i)-(round((img\_one(i)-d)/2/delta)\*2\*delta+d)); %Q(+1) % pdist默认欧氏距离 euclidean

b2 = norm(img\_one(i)-(round((img\_one(i)-d-delta)/2/delta)\*2\*delta+d+delta));%Q(-1)

if b1 < b2

temp=[temp,'0'];

if mod(i,8)==0%每八位获取一个字符

watermark=[watermark,char(bin2dec(temp))];

temp='';

end

else

temp=[temp,'1'];

if mod(i,8)==0

watermark=[watermark,char(bin2dec(temp))];

temp='';

end

end

end

disp("您所嵌入的水印信息为：")

disp(watermark)

I\_double=double(img\_yuan);

J1=img;

J1\_double=double(J1);

D=J1\_double-I\_double;%PSNR计算公式

MSE = sum(D(:).\*D(:)) / numel(img); %numel计算数组中的元素个数

PSNR = 10\*log10(255^2 / MSE);

disp("PSNR为:")

disp(PSNR)

if PSNR>40

disp('图像质量极好，非常接近原图')

end

if PSNR <=40

if PSNR>30

disp('图像失真肉眼可见但是可以接受')

end

end

if PSNR <=30

if PSNR>20

disp('图像质量差')

end

end

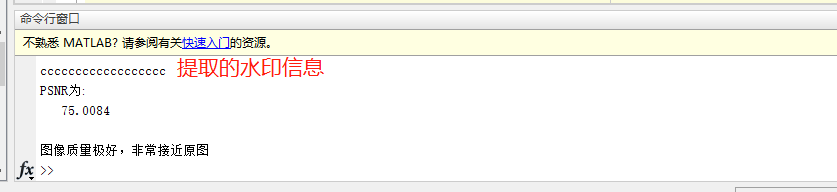
if PSNR <=20

disp('图像不可接受')

end

1. 实验结果

显示提取出嵌入的水印信息，并显示PSNR值



3.2 评价水印算法

1. PSNR
2. 通过嵌入不同容量的水印信息，并计算不同容量的PSNR值，画成曲线图，查看特点，本次实验测试图像：50幅
3. 代码

for i=1:50

J1=imread(['lena\_watermark',sprintf('%01d',i),'.bmp']);

J1\_double=double(J1);

D=J1\_double-I\_double;%PSNR计算公式

MSE = sum(D(:).\*D(:)) / numel(img\_yuan); %numel计算数组中的元素个数

PSNR(i) = 10\*log10(255^2 / MSE);

x(i)=i;

end

figure;

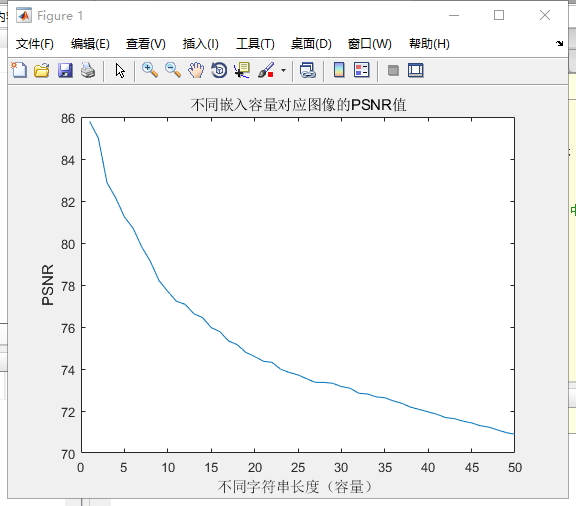
plot(x,PSNR)%画出PSNR图像

xlabel('不同字符串长度（容量）')

ylabel('PSNR')

title('不同嵌入容量对应图像的PSNR值')

3.实验结果



1. 实验分析：我们发现随着嵌入的水印长度越来越长，图像的PSNR值逐渐在减小，说明图像的质量在减小，且随着长度的增加，PSNR减小的幅度在缩小
2. 正确性
3. 正检率计算

function strsim = mystrsim(target,source) %X为字符串

len1=length(source);

len2=length(target);

d=zeros(len1+1,len2+1);

for i=2:length(source)+1

d(i,1)= i;

end

for j=2:length(target)+1

d(1,j)= j;

end

for i =2:length(source)+1

for j = 2:length(target)+1

if source(i - 1) == target(j - 1)

d(i,j) = d(i-1,j-1); %不需要编辑操作

else

edIns = d(i,j-1)+1; %source 插入字符

edDel = d(i-1,j)+1; %source 删除字符

edRep = d(i-1,j-1)+1; %source 替换字符

d(i,j)= min(min(edIns, edDel),edRep);

end

end

end

y=d(length(source)+1,length(target)+1); %最少操作次数

strsim=1/(y+1); %相似度

end

2.根据实验可知，只要嵌入水印的长度不超过最大值，水印提取都是100%的正确性，说明正确性与嵌入水印的长度无关，实验代码及结果如下

实验代码：

for i=1:50

J1=imread(['lena\_watermark',sprintf('%01d',i),'.bmp']);

J1\_double=double(J1);

D=J1\_double-I\_double;%PSNR计算公式

MSE = sum(D(:).\*D(:)) / numel(img\_yuan); %numel计算数组中的元素个数

PSNR(i) = 10\*log10(255^2 / MSE);

x(i)=i;

end

success=[];

zhengque=[];

cuowu=[];

for i=1:50

J1=imread(['lena\_watermark',sprintf('%01d',i),'.bmp']);

F=J1;

%F = imnoise(J1,'speckle',0.01);

[I\_x, I\_y] = size(F);

n=I\_x\*I\_y;

imgg=F';

img\_one=double(imgg(:));%变为一维数组

origin='';

leng = '';%提取水印长度

% 量化器

% Q1从0开始，Q2从d开始，步长为delta

d = 5;

delta = 3;

Q1 = 0:2\*delta:n; % -1

Q2 = d:2\*delta:n; % 1

for k=1:8

b1 = norm(img\_one(k)-(round((img\_one(k)-d)/2/delta)\*2\*delta+d)); % pdist默认欧氏距离 euclidean

b2 = norm(img\_one(k)-(round((img\_one(k)-d-delta)/2/delta)\*2\*delta+d+delta));

if b1 < b2

leng=[leng,'0'];

else

leng=[leng,'1'];

end

end

num=bin2dec(leng);

cha\_num=num\*8+8;%根据长度提取信息

watermark='';

temp='';

for k=9:cha\_num%真正开始提取水印

b1 = norm(img\_one(k)-(round((img\_one(k)-d)/2/delta)\*2\*delta+d)); %Q(+1) % pdist默认欧氏距离 euclidean

b2 = norm(img\_one(k)-(round((img\_one(k)-d-delta)/2/delta)\*2\*delta+d+delta));%Q(-1)

if b1 < b2

temp=[temp,'0'];

if mod(k,8)==0 %每八位获取一个字符

watermark=[watermark,char(bin2dec(temp))];

temp='';

end

else

temp=[temp,'1'];

if mod(k,8)==0

watermark=[watermark,char(bin2dec(temp))];

temp='';

end

end

end

for j=1:i

origin=[origin,'a'];

end

zhengque(i)=0;

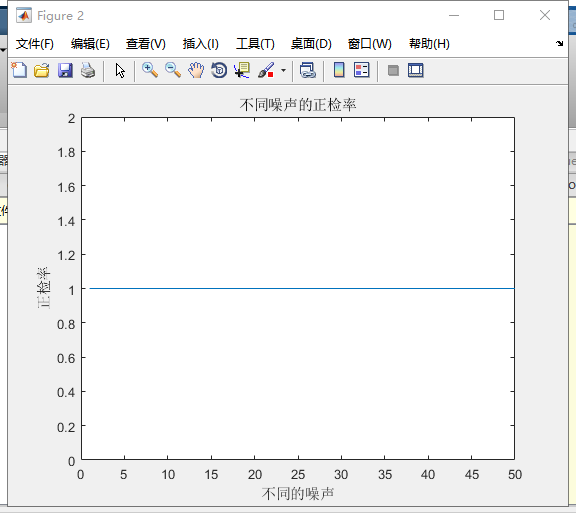
cuowu(i)=0;

success(i)=0.0;

success(i)=mystrsim(origin,watermark);%调用正检率函数

end

实验结果



3.对同一掩密载体（嵌入了50个字符出的图像）进行加噪处理，对其亮暗做处理，进行不同的亮暗变换， 即像素值进行加减，得到不同的正检率

加噪代码：（对每个像素进行加不同的值）

distance=double(i/5);

for sj=1:length(img\_one)

img\_one(sj)=img\_one(sj)+distance;

end

画图代码：

figure;

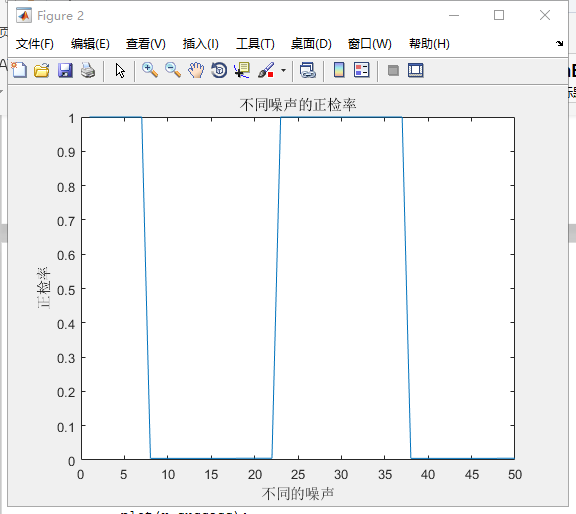
plot(x,success);

xlabel('不同的噪声');

ylabel('正检率');

title('不同噪声的正检率');

实验结果：



实验分析：在不同的亮度噪声下，正检率改变较大，说明QIM针对亮度处理的鲁棒性不是很强

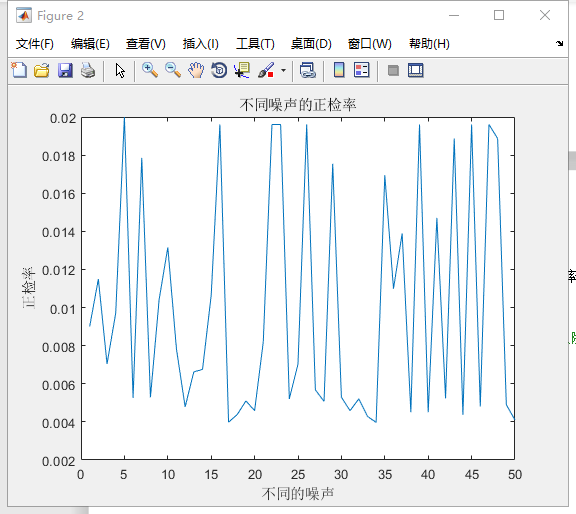
1. 随机噪声：对同一掩密载体加入随机噪声，并查看正检率

加噪代码：

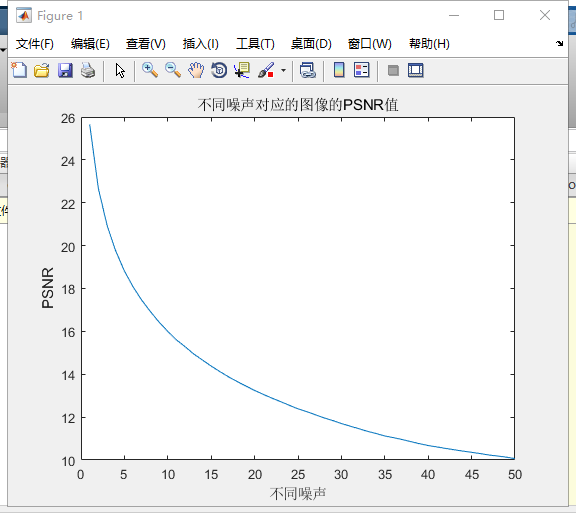
can=i/100;

F = imnoise(J1,'speckle',can); %根据参数加入随机噪声

实验结果



加噪后的PSNR图：



实验分析：图像加噪以后PSNR值非常小，图像质量会变得很差，图像提取水印的正确率非常差，都在2%以下，这说明QIM抵抗不了噪声的攻击。