**《数字内容安全》**

**实验报告**

**实验名称：** 变换域水印嵌入和提取 **任课教师：付宇**

**学号：**200340164 **姓名：** 张博文  **提交日期：** 2022/11/7

**一、实验目标：**

实现基于 DCT/DWT 的水印嵌入和提取

**二、实验设计：**

1、W-SVD 数字水印的生成和嵌入

编写函数 wavemarksvd. m 完成 W-SVD 数字水印算法的实验。

2、W-SVD 数字水印检测

编写函数 wavedetect. m 完成检测实验。

**三、实验记录：**

1、W-SVD 数字水印的生成和嵌入

% 函数功能 : 本函数将完成 W-SVD 模型下数字水印的嵌入

% 输入格式举例 :

% [ watermarkimagergb, watermarkimage, waterCA, watermark, correlationU, correlationV]

% = wavemarksvd( 'c: \ lenna. jpg', 'c: \test. png', 1983, 'db6', 2, 0 . 1 , 0. 99 )

% 参数说明 :

% input 为输入原始图像

% seed 为随机数种子

% wavelet 为使用的小波函数

% level 为小波分解的尺度

% alpha 为水印强度

% ratio 为算法中 d /n 的比例

% watermarkimagergb 为加有水印的结果

% watermarkimage 为单层加水印的结果

% waterCA 为加有水印模板的低频分解系数

% watermark2 为由水印模板直接重构得到的水印形态 , 仅便于直观认识 , 本身无意义。

% correlationU, correlationV 为替换正交矩阵后与未替换的正交矩阵的相关系数

function [ watermarkimagergb, watermarkimage, waterCA, watermark2, correlationU, correlationV] = wavemarksvd( input, goal, seed, wavelet, level, alpha, ratio)

% function watermark = wavemarksvd( input, goal, seed, wavelet, level, alpha, ratio)

% 读取原始图像

data = imread( input) ;

data = double( data) /255;

datared = data(:,:, 1) ; % 在 R 层加水印

% 对原始图像的 R 层进行小波分解记录原始大小 , 并将其补成正方形

[ C, Sreal] = wavedec2 ( datared, level, wavelet) ;

[ row, list] = size( datared) ;

standard1 = max( row, list) ;

new = zeros( standard1, standard1) ;

if row <= list

new( 1:row,:) = datared;

else

new(:, 1:list) = datared;

end

% 正式开始加水印

% 小波分解 , 提取低频系数

[ C, S] = wavedec2 ( new, level, wavelet) ;

CA = appcoef2( C, S, wavelet, level) ;

% 对低频系数进行归一化处理

[ M, N] = size( CA) ;

CAmin = min( min( CA) ) ;

CAmax = max( max( CA) ) ;

CA = ( 1 /( CAmax - CAmin) ) \* ( CA - CAmin) ;

d = max( size( CA) ) ;

% 对低频率系数单值分解

[ U, sigma, V] = svd( CA) ;

% 按输出参数得到要替换的系数的数量

np = round( d\* ratio) ;

% 以下是随机正交矩阵的生成

rand( 'seed', seed) ;

M\_V = rand( d, np) - 0.5;

[ Q\_V, R\_V] = qr( M\_V, 0) ;

M\_U = rand( d, np) - 0.5;

[ Q\_U, R\_U] = qr( M\_U, 0) ;

% 替换

V2 = V; U2 = U;

V(:, d - np + 1:d) = Q\_V(:, 1: np) ;

U(:, d - np + 1:d) = Q\_U(:, 1: np) ;

sigma\_tilda = alpha\* flipud( sort( rand( d, 1) ) ) ;

correlationU = corr2( U, U2) ; % 计算替换的相关系数

correlationV = corr2( V, V2) ;

% 生成水印

watermark = U\* diag( sigma\_tilda, 0) \* V';

% 重构生成水印的形状 , 便于直观认识 , 本身无意义

watermark2 = reshape( watermark, 1, S( 1, 1) \* S( 1, 2) ) ;

waterC = C;

waterC( 1, 1: S( 1, 1) \* S( 1, 2) ) = watermark2;

watermark2 = waverec2 ( waterC, S, wavelet) ;

% 调整系数生成嵌入水印后的图像

CA\_tilda = CA + watermark;

over1 = find( CA\_tilda > 1) ;

below0 = find( CA\_tilda < 0) ;

CA\_tilda( over1) = 1;

CA\_tilda( below0 ) = 0; % 系数调整 , 将过幅系数与负数 修正

CA\_tilda = ( CAmax-CAmin) \* CA\_tilda +CAmin; %系数还原到归一化以前的范围

% 记录加有水印的低频系数

waterCA = CA\_tilda;

if row <= list

waterCA = waterCA(1:Sreal( 1, 1),:);

else

waterCA = waterCA(:, 1: Sreal( 1, 2) ) ;

end

% 重构

CA\_tilda = reshape( CA\_tilda, 1, S( 1, 1) \* S( 1 , 2) ) ;

C( 1, 1:S( 1, 1) \* S( 1, 2) ) = CA\_tilda;

watermarkimage = waverec2( C, S, wavelet) ;

% 将前面补上的边缘去掉

if row <= list

watermarkimage = watermarkimage( 1:row,:) ;

else

watermarkimage = watermarkimage(:, 1:list) ;

end

watermarkimagergb = data;

watermarkimagergb(:,:, 1) = watermarkimage;

imwrite( watermarkimagergb, goal, 'BitDepth', 16) ; % 通过写回修正过幅系数

watermarkimagergb2 = imread( goal) ;

figure( 1) ;

subplot( 321) ; imshow( watermark2\* 255) ; title( '水印形态图') ;

subplot( 323) ; imshow( data) ; title( '原始图像 ') ;

subplot( 324) ; imshow( watermarkimagergb2 ) ; title( '嵌入水印后的 RGB 图像') ;

subplot( 325) ; imshow( datared) ; title( 'R 层图像') ;

subplot( 326) ; imshow( watermarkimage) ; title('嵌入水印后的 R 层图像') ;

执行结果：

QR 代码

描述已自动生成

2、W-SVD 数字水印检测

编写函数 wavedetect. m 完成检测实验。

% 函数功能 : 本函数将完成 W-SVD 模型下数字水印的检测

% 输入格式举例 :

% [ corr\_coef, corr\_DCTcoef ] = wavedetect ( 'c: \ test. png', 'c: \ lenna. jpg', 1983,'db6', 2 , 0.1, 0.99)

% input 为输入原始图像

% seed 为随机数种子

% wavelet 为使用的小波函数

% level 为小波分解的尺度

% alpha 为水印强度

% ratio 为算法中 d /n 的比例

% corr\_coef, corr\_DCTcoef 分别为不同方法下检测出的相关性值

function [ corr\_coef, corr\_DCTcoef] = wavedetect( test, original, seed, wavelet, level,alpha, ratio)

dataoriginal = imread( original) ;

datatest = imread( test) ;

dataoriginal = double( dataoriginal) /255;

datatest = double( datatest) /65535;

% 请大家注意这里的两个分母 , 这是与图像文件格式有关的

dataoriginal = dataoriginal(:,:, 1) ;

datatest = datatest(:,:,1 ) ;

% 提取加有水印的图像的小波低频系数

[ watermarkimagergb, watermarkimage, waterCA, watermark2, correlationU, correlationV]= wavemarksvd( original, test, seed, wavelet, level, alpha, ratio);

% 提取待测图像的小波低频系数

[ C, S] = wavedec2 ( datatest, level, wavelet) ;

CA\_test = appcoef2( C, S, wavelet, level) ;

% 提取原始图像的小波低频系数

[ C, S] = wavedec2 ( dataoriginal, level, wavelet) ;

realCA = appcoef2( C, S, wavelet, level) ;

% 生成两种水印

realwatermark = waterCA-realCA;

testwatermark = CA\_test-realCA;

% 计算相关性值

corr\_coef = trace ( realwatermark'\* testwatermark) / ( norm( realwatermark,'fro') \* norm( testwatermark, 'fro') ) ;

% DCT 系数比较

DCTrealwatermark = dct2( waterCA-realCA) ;

DCTtestwatermark = dct2 ( CA\_test-realCA) ;

DCTrealwatermark = DCTrealwatermark ( 1: min ( 32 , max ( size ( DCTrealwatermark)) ) , 1 :min( 32, max( size( DCTrealwatermark) ) ) ) ;

DCTtestwatermark = DCTtestwatermark ( 1:min ( 32, max ( size ( DCTtestwatermark)) ) , 1:min( 32, max( size( DCTtestwatermark) ) ) ) ;

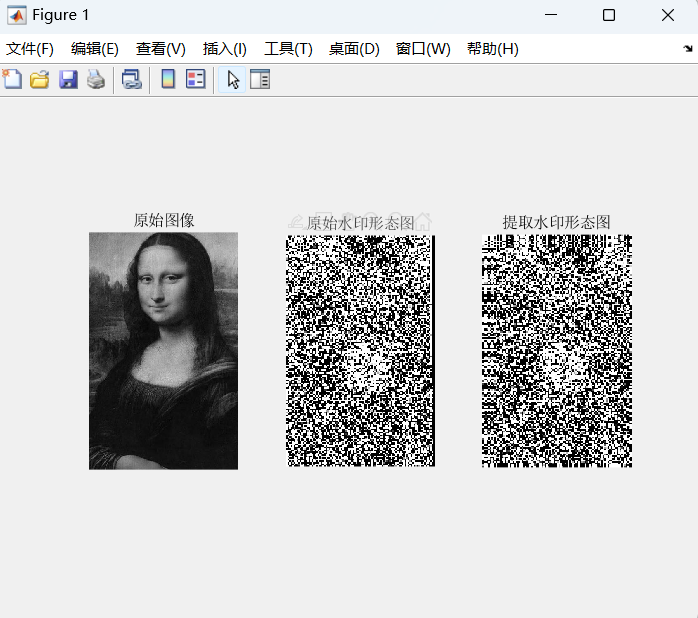
DCTrealwatermark( 1, 1) = 0;

DCTtestwatermark( 1, 1) = 0;

corr\_DCTcoef = trace ( DCTrealwatermark'\* DCTtestwatermark) /( norm( DCTrealwatermark,'fro') \* norm( DCTtestwatermark, 'fro') ) ;

**图形用户界面, 应用程序

描述已自动生成**

****

**四、实验思考与体会**

本此实验提出了一种基于dct域的Matlab水印算法，实验结果表 明这种算法具有较好的性能，实现的水印具有不可见性,而且具有较好的鲁棒性。

实验发现一开始图片选用.jpg，使用wavemarksvd函数报错：错误使用 writejpg (第 59 行)仅灰度图像支持 16 位数据。原因是JPEG和PNG等格式才有BitDepth，后来改成选用PNG图片才解决。

老师的代码wavedetect.m中在调用wavemarksvd函数时传入的第二个参数写错了