——————————————————————————**装 订 线**————————————————————————————————**—**

**西安邮电大学**

**通信与信息工程学院**

**《图像与视频处理实验》**

**报告书**

**专业班级**：广电1802

**学生姓名**：李世钰

**学 号**： 03185043

**班内序号**： 05

实验五 图像分割实验

一、实验目的

(1) 掌握基于灰度阈值化的图像分割基本原理；

(2) 掌握基于边缘检测的图像分割基本原理；

(3) 掌握基于区域的图像分割基本原理。

二、实验内容

(1) 使用最大类间方差法，实现图像分割；

(2) 使用canny算法，实现图像边缘检测与图像分割；

(3) 使用区域生长法实现图像分割。

三、实验要求

1、使用最大类间方差法，实现图像分割

(1) 给定图像素材：noisy\_fingerprint.tif

(2) 使用graythresh()函数计算最佳分割门限；

(3) 使用im2bw()函数，实现图像二值化，完成图像分割。

2、边缘检测与目标外边缘追踪

(1) 给定图像素材为：coins.png;

(2) 使用edge()函数，采用“canny”算子，选择合适的参数检测边缘。

(3) 使用imfill()函数，填充边界内部孔洞；

(4) 使用bwboundaries()对填充后的二值图像进行外边缘追踪，完成图像跟踪。

(5) 在原图像coins.png中以不同颜色显示追踪到的外边界。

3、区域生长

(1) 给定图像素材coins.png

(2) 参考教材Page120，编写基于区域生长的图像分割程序。

(3) 实验报告中以图示方式给出选择的种子点的坐标位置。

四、实验原理

**MATLAB函数说明**

graythresh()函数

功能：使用Otsu算法计算灰度图像最佳分割门限

调用格式：level = graythresh(I)；

输入参数：I为灰度图像

输出参数：level为Otsu算法计算出的归一化([0 1])最佳阈值

im2bw()函数

功能：阈值化处理，实现图像的二值化

调用格式：BW = im2bw(I, th)；

输入参数：I为灰度图像；th为阈值

输出参数：BW阈值化处理后的二值图像

graythresh()函数举例

I = imread(‘noisy\_fingerprint.tif’);

th = graythresh(I);

th = th\*255

edge()

功能：目标边缘检测

调用格式: [BW,T] = edge(I,’sobel’,thresh, direction)

[BW,T] = edge(I,’prewitt’,thresh,direction)

[BW,T] = edge(I,’roberts’,thresh,direction)

[BW,T] = edge(I, ’log’, thresh, sigma)

[BW,T] = edge(I, ’canny’, thresh, sigma)

输入参数：

I：灰度图像；

‘sobel’, ‘prewitt’, ‘roberts’ ,’log’ ,’canny’表示检测算子

输入参数：

direction：指定方向可选’vertical’, ’horizontal’,’ both’

thresh： 指定阈值门限。注：除’canny’算子外， thresh 均为标量；’canny’算子是双阈值，故thresh可指定为向量 [thresh1 thresh2]，且 thresh1 < thresh2

sigma： 指定’LoG’和’canny’算子的高斯滤波器的标准差

输出参数：

BW： 二值图像；

T ： 门限。

imfill()

功能： 填充区域和孔洞

调用格式：

BW2 = imfill(BW,location);

BW2 = imfill(BW,’holes’);

输入参数：

location ：指定开始填充的坐标；

bwboundaries()

功能：追踪区域边界(包括外边界和孔洞边界)

调用格式： [B, L, N, A] = bwboundaries(BW, conn, option)；

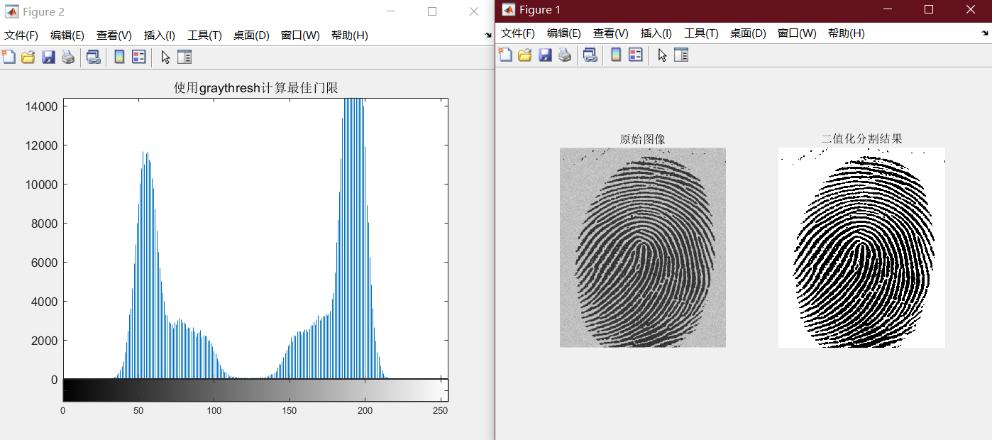
输入参数：

BW：灰度图像；

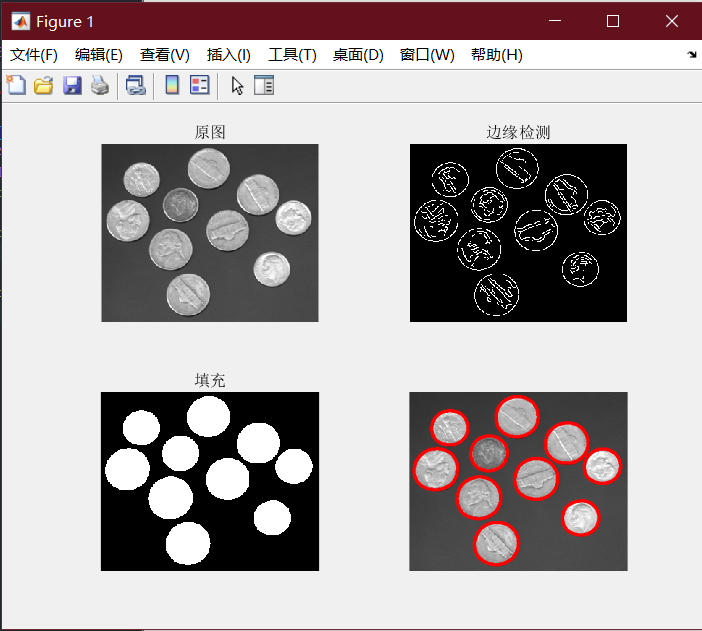
conn：指定连通性，当追踪子边界和父边界时使用；

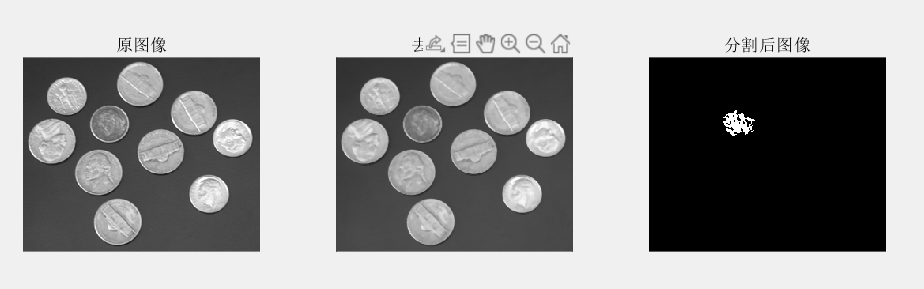
option：可选’holes’或’noholes’指定是否追踪孔洞边界；

五、实验结果与分析

**使用最大类间方差法，实现图像分割**

**边缘检测与目标外边缘追踪**

****

**区域生长**

**附程序源码：**

**noisy\_fingerprint.m**

clear;

clc;

I = imread('noisy\_fingerprint.tif');

th = graythresh(I);

J = im2bw(I, th);

th = 255 \* th;

subplot(1, 2, 1);

imshow(I); title('原始图像');

subplot(1, 2, 2);

imshow(J); title('二值化分割结果');

figure, imhist(I);

title('使用graythresh计算最佳门限');

**coins.m**

clear;

clc;

I = imread('coins.png');

I1 = edge(I, 'canny', [0.08 0.16]);

I2 = imfill(I1, 'holes');

figure;

subplot(2,2,1);

imshow(I);

title('原图');

subplot(2,2,2);

imshow(I1);

title('边缘检测');

subplot(2,2,3);

imshow(I2);

title('填充');

[B, L, N, A1] = bwboundaries(I2);

subplot(2,2,4);imshow(I); hold on;

    for k = 1:length(B)

        boundary = B{k};

        if(k > N)

            plot(boundary(:,2),...

                boundary(:,1),'g','LineWidth',2);

        else

            plot(boundary(:,2),...

                boundary(:,1),'r','LineWidth',2);

        end

    end

hold off

**grow.m**

clc;

clear;

A = imread('coins.png');

B = im2gray(A);

C = medfilt2(B, [3, 3]);

figure(1);

subplot(131); imshow(A); title('原图像');

subplot(132); imshow(C); title('去噪后图像');

I = double(C);

[M, N] = size(I);

J = zeros(M, N);

msgbox('鼠标单击选择一种子点后按回车键', '操作', 'modal');

[y, x] = getpts;

xr = round(x);

yr = round(y);

seed = I(xr, yr);

J(xr, yr) = 1;

graysum = seed;

pointsum = 1;

count = 1;

threshold = 13;

while count > 0

    s = 0;

    count = 0;

    for i = 1:M

        for j = 1:N

            if J(i, j) == 1

                if (i - 1) > 0 && (i + 1) < (M + 1) && (j - 1) > 0 && (j + 1) < (N + 1)

                    for u = -1:1

                        for v = -1:1

                            if J(i + u, j + v) == 0 && abs(I(i + u, j + v) - seed) <= threshold

                                J(i + u, j + v) = 1;

                                count = count + 1;

                                s = s + I(i + u, j + v);

                            end

                        end

                    end

                end

            end

        end

    end

    pointsum = pointsum + count;

    graysum = graysum + s;

    seed = graysum / pointsum;

end

subplot(133), imshow(J);

title('分割后图像');