

PROJECT 10-02 [Multiple Uses]

Global Thresholding

- (a) Write a global thresholding program in which the threshold is estimated automatically using the procedure discussed in Section 10.3.2. The output of your program should be a segmented (binary) image.
- (b) Download Fig. 10.38(a) and apply your program to it. The result should be the same as in Example 10.15.

一、实验用图

本项目中，采用书中图 10.38 (a) 中大小为 958×798 像素的灰度图像。

二、实验过程

- 1、首先读取一幅图像 10.38 (a)，然后输出显示这幅原始图像
- 2、统计图像的直方图，并输出这幅直方图
- 3、对原始图像进行长宽统计，并计算出初始全局阈值
- 4、进行全局阈值的迭代运算
- 5、使用全局阈值进行图像分割，并输出阈值处理图像

三、程序源代码

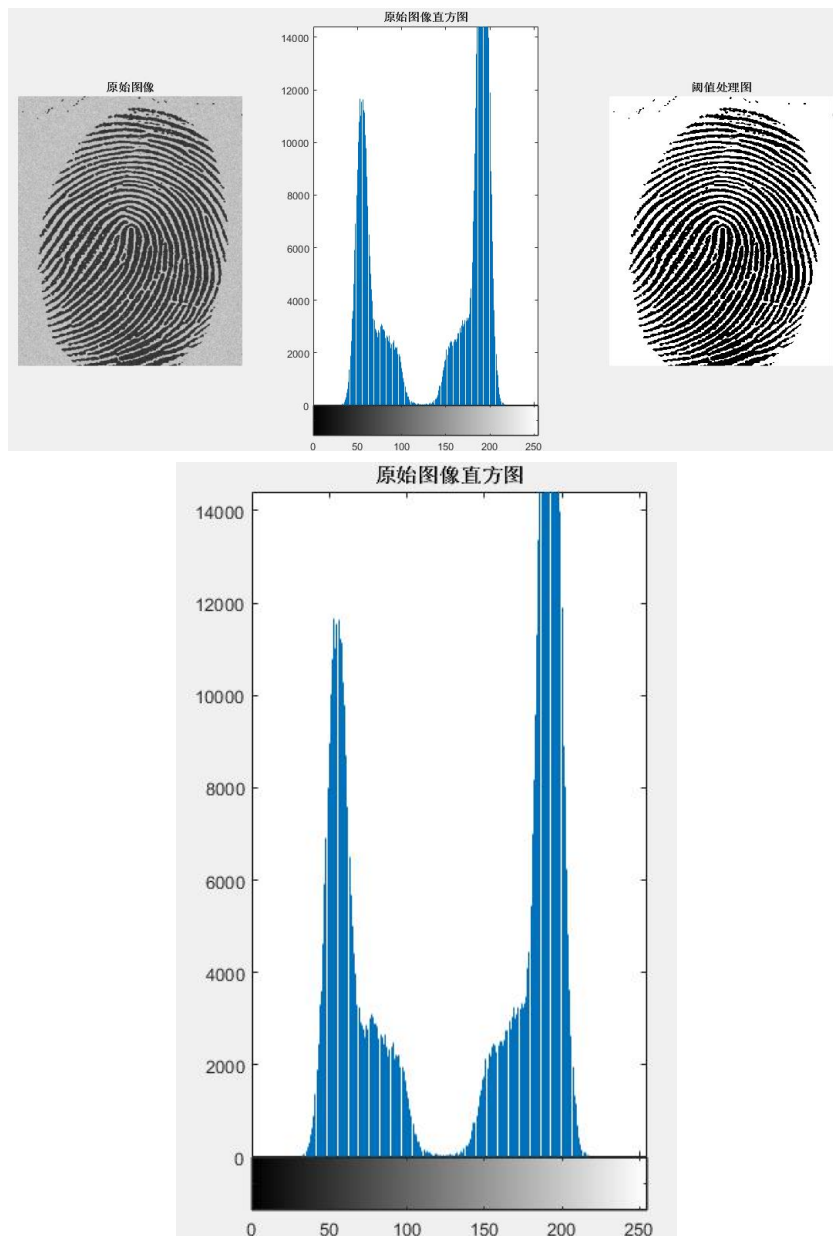
```
1 -   clc
2 -   clear
3
4 -   A=imread('Fig1038(a).tif');%读入原始图像
5 -   subplot(1,3,1);
6 -   imshow(A);%显示原始图像
7 -   title('原始图像');
8 -   subplot(1,3,2)
9 -   imhist(A)%显示原始图像直方图
10 -  title('原始图像直方图');
11
12 -  [height,width] = size(A);%测出原始图像的长宽
13 -  I=sum(sum(double(A)))/(height*width);%计算初始全局阈值
```

```

15 -     done=false;
16 -     while ~done
17 -         Inext=0.5*(mean(A(A>=I))+mean(A(A<I))):%I的迭代运算
18 -         done=abs(I-Inext)<0.01;
19 -         I=Inext;
20 -     end
21
22 -     B=im2bw(A, I/255):%使用得到的I值分割图像
23 -     subplot(1,3,3)
24 -     imshow(B)%显示阈值处理结果图
25 -     title('阈值处理图');

```

四、程序运行结果



原始图像



阈值处理图



五、结果分析

由于阈值处理直观、实现简单且计算速度快，因此图像阈值处理在图像分割应用中处于核心地位。而当物体和背景像素的灰度分布十分明显时，全局阈值处理可以取得十分出色的结果。

全局阈值处理中有一个迭代算法用来估计阈值，这种算法在物体和背景相关的直方图模式间存在一个相当清晰的波谷时，这个算法可以工作的很好。不过当图像中存在较为严重的噪声时，这个算法的结果就不是非常理想。

在本次实验中，从原始图像的直方图中可以看到。原始图像中不存在严重的噪声，而且物体和背景间存在一个相当清晰的波谷。所以全局阈值处理可以取得非常理想的结果，而且阈值的迭代算法中迭代次数也非常少。可以很快速的得出结果。