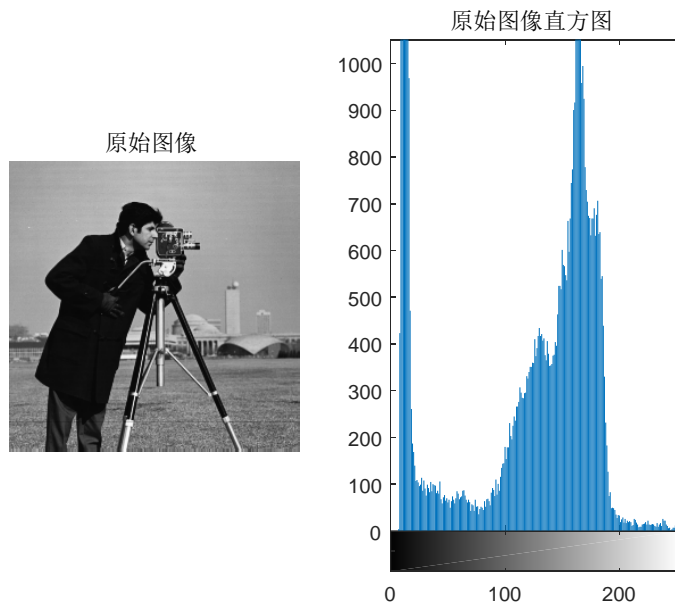


# 数字图像处理

## 1. 图像读取及灰度变换

```
I=imread('cameraman.tif');%读取图像
subplot(1,2,1),imshow(I) %输出图像
title('原始图像') %在原始图像中加标题
subplot(1,2,2),imhist(I) %输出原图直方图
title('原始图像直方图') %在原图直方图上加标题
```



## 2. 图像旋转

```
I = imread('cameraman.tif');
figure,imshow(I);
theta = 30;
K = imrotate(I,theta); % Try varying the angle, theta.
figure, imshow(K)
```



### 3.图像反转

MATLAB 程序实现如下：

```
I=imread('boats.bmp');
J=double(I);
J=-J+(255-1);          %图像反转线性变换
H=uint8(J);
subplot(1,2,1),imshow(I);
subplot(1,2,2),imshow(H);
```

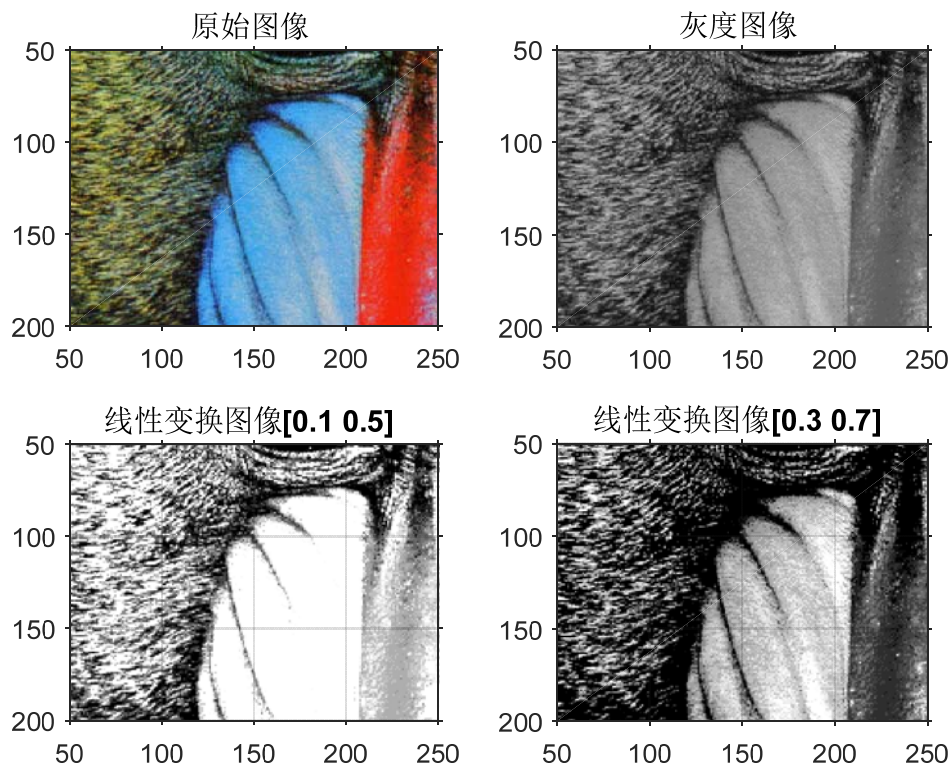


### 4.灰度线性变换

MATLAB 程序实现如下：

```
I=imread('baboon.bmp');
subplot(2,2,1),imshow(I);
title('原始图像');
axis([50,250,50,200]);
axis on;          %显示坐标系
I1=rgb2gray(I);   %转为灰度图像
subplot(2,2,2),imshow(I1);
title('灰度图像');
axis([50,250,50,200]);
axis on;          %显示坐标系
J=imadjust(I1,[0.1 0.5],[]); %局部拉伸，把[0.1 0.5]内的灰度拉伸为[0 1]
% J = imadjust(I,[low_in; high_in],[low_out; high_out],gamma)
%将图像 I 中的亮度值映射到 J 中的新值，即将 low_in 至 high_in 之间的值映射到 low_out
%至 high_out 之间的值。low_in 以下与 high_in 以上的值被剪切掉了，也就是说，low_in 以下
%的值映射到 low_out，high_in 以上的值映射到 high_out。它们都可以使用空的矩阵[]，默认
%值是[0 1]。
subplot(2,2,3),imshow(J);
title('线性变换图像[0.1 0.5]'); %对比度
axis([50,250,50,200]);
grid on;          %显示网格线
axis on;          %显示坐标系
K=imadjust(I1,[0.3 0.7],[]); %局部拉伸，把[0.3 0.7]内的灰度拉伸为[0 1]
subplot(2,2,4),imshow(K);
title('线性变换图像[0.3 0.7]');
axis([50,250,50,200]);
```

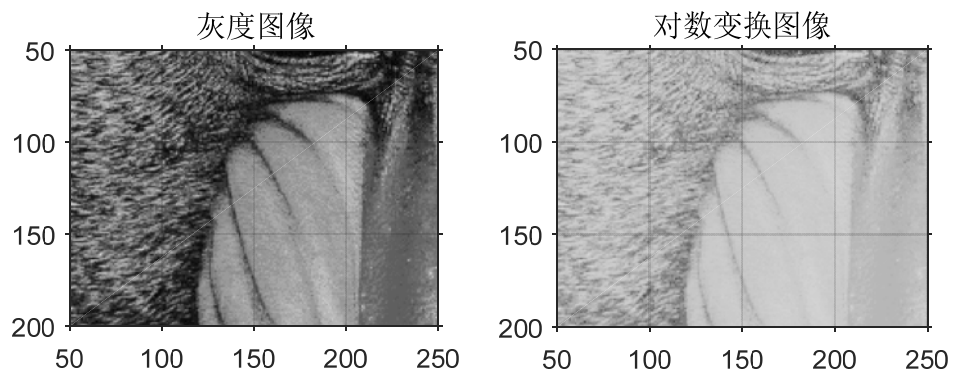
```
grid on;           %显示网格线
axis on;           %显示坐标系
```



## 5.非线性变换

MATLAB 程序实现如下:

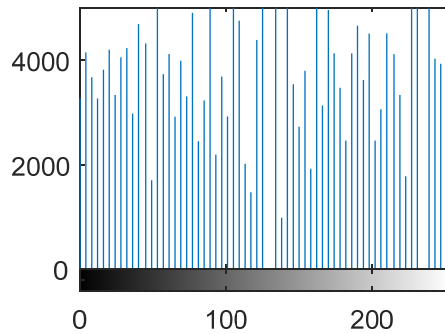
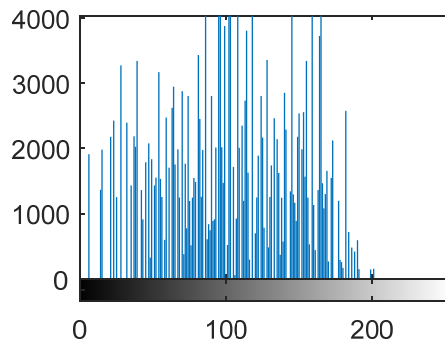
```
I=imread('baboon.bmp ');
I1=rgb2gray(I);
subplot(1,2,1),imshow(I1);
title('灰度图像');
axis([50,250,50,200]);
grid on;           %显示网格线
axis on;           %显示坐标系
J=double(I1);
J=40*(log(J+1));
H=uint8(J);
subplot(1,2,2),imshow(H);
title('对数变换图像');
axis([50,250,50,200]);
grid on;           %显示网格线
axis on;           %显示坐标系
```



## 6.直方图均衡化

MATLAB 程序实现如下：

```
I=imread('baboon.bmp');
I=rgb2gray(I);
figure;
subplot(2,2,1);
imshow(I);
subplot(2,2,2);
imhist(I); %提取图像中的直方图信息
I1=histeq(I); %直方图均衡化
figure;
subplot(2,2,1);
imshow(I1);
subplot(2,2,2);
imhist(I1);
```



## 7. 均值滤波器（用均值替代原图像中的各个像素值）

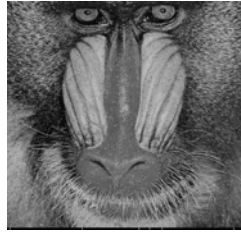
用 MATLAB 实现领域平均法抑制噪声程序：

```
I=imread('baboon.bmp');
subplot(2,3,1)
imshow(I)
title('原始图像')
I=rgb2gray(I);
I1=imnoise(I,'salt & pepper',0.02); %添加噪声污染一幅图，椒盐噪声
subplot(2,3,2)
imshow(I1)
title('添加椒盐噪声的图像')
k1=filter2(fspecial('average',3),I1)/255; %进行 3*3 模板平滑滤波
% h = fspecial(type,parameters) 'average', 为均值滤波，参数为 n，代表模版尺寸。
k2=filter2(fspecial('average',5),I1)/255; %进行 5*5 模板平滑滤波
k3=filter2(fspecial('average',7),I1)/255; %进行 7*7 模板平滑滤波
k4=filter2(fspecial('average',9),I1)/255; %进行 9*9 模板平滑滤波
subplot(2,3,3),imshow(k1);title('3*3 模板平滑滤波');
subplot(2,3,4),imshow(k2);title('5*5 模板平滑滤波');
subplot(2,3,5),imshow(k3);title('7*7 模板平滑滤波');
subplot(2,3,6),imshow(k4);title('9*9 模板平滑滤波');
```

原始图像



添加椒盐噪声的图像



3\*3 模板平滑滤波



5\*5 模板平滑滤波



7\*7 模板平滑滤波



9\*9 模板平滑滤波



**8. 中值滤波器**(按灰度值进行排序,然后选择该序列的中间值作为输出的像素值)

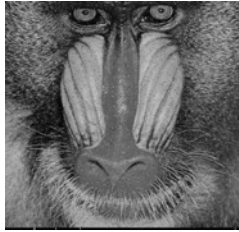
用 MATLAB 实现中值滤波程序如下:

```
I=imread('baboon.bmp');
I=rgb2gray(I);
J=imnoise(I,'salt&pepper',0.02);
subplot(231),imshow(I);title('原图像');
subplot(232),imshow(J);title('添加椒盐噪声图像');
k1=medfilt2(J);           %进行 3*3 模板中值滤波
k2=medfilt2(J,[5,5]);     %进行 5*5 模板中值滤波
k3=medfilt2(J,[7,7]);     %进行 7*7 模板中值滤波
k4=medfilt2(J,[9,9]);     %进行 9*9 模板中值滤波
subplot(233),imshow(k1);title('3*3 模板中值滤波');
subplot(234),imshow(k2);title('5*5 模板中值滤波 ');
subplot(235),imshow(k3);title('7*7 模板中值滤波');
subplot(236),imshow(k4);title('9*9 模板中值滤波');
```

原图像



添加椒盐噪声图像



3\*3模板中值滤波



5\*5模板中值滤波



7\*7模板中值滤波

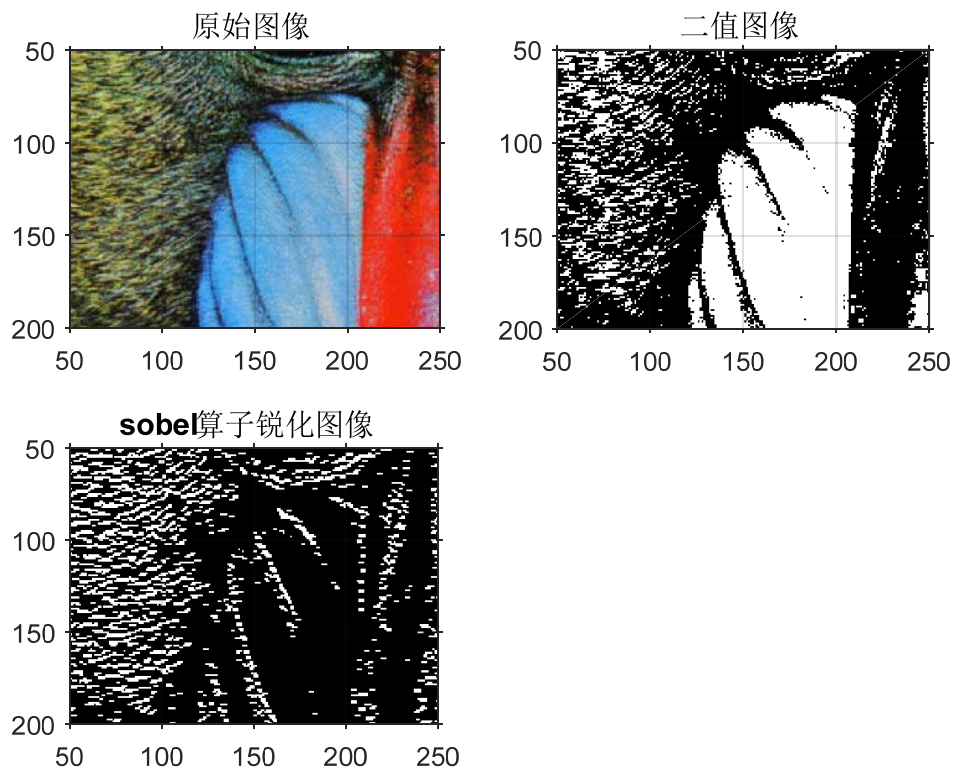


9\*9 模板中值滤波



### 9.用 Sobel 算子图像锐化:

```
I=imread('baboon.bmp');  
subplot(2,2,1),imshow(I);  
title('原始图像');  
axis([50,250,50,200]);  
grid on; %显示网格线  
axis on; %显示坐标系  
I1=im2bw(I);  
subplot(2,2,2),imshow(I1);  
title('二值图像');  
axis([50,250,50,200]);  
grid on; %显示网格线  
axis on; %显示坐标系  
H=fspecial('sobel'); %选择 sobel 算子  
J=filter2(H,I1); %卷积运算  
subplot(2,2,3),imshow(J);  
title('sobel 算子锐化图像');  
axis([50,250,50,200]);  
grid on; %显示网格线  
axis on; %显示坐标系
```



## 10.梯度算子检测边缘

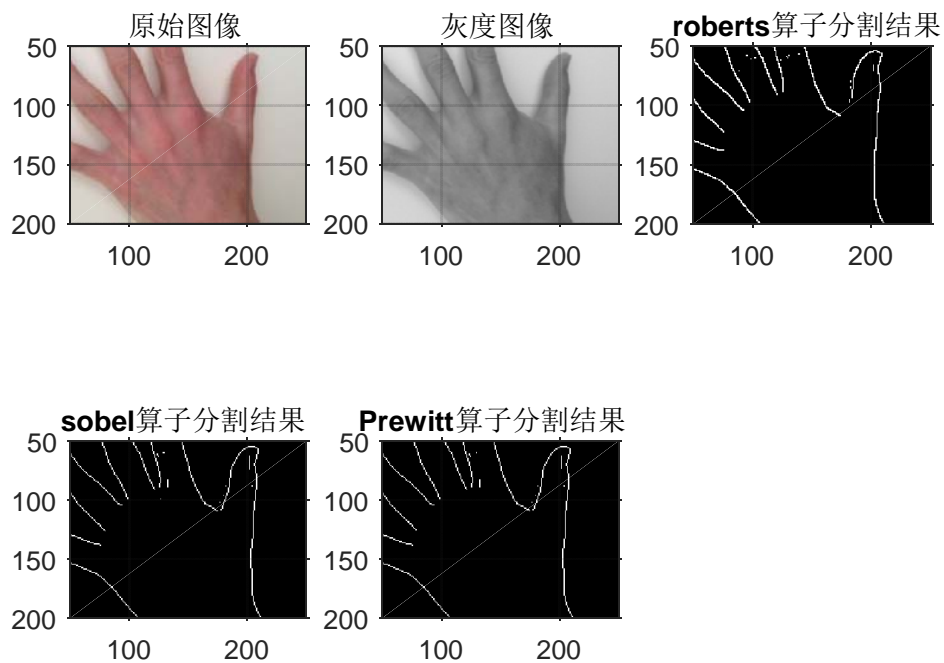
```
I=imread('hands1.jpg');
subplot(2,3,1);
imshow(I);
title('原始图像');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系
I1=rgb2gray(I);
subplot(2,3,2);
imshow(I1);
title('灰度图像');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系
I2=edge(I1,'roberts');
subplot(2,3,3);
imshow(I2);
title('roberts 算子分割结果');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系
I3=edge(I1,'sobel');
subplot(2,3,4);
```



```

imshow(I3);
title('sobel 算子分割结果');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系
I4=edge(I1,'Prewitt');
subplot(2,3,5);
imshow(I4);
title('Prewitt 算子分割结果 ');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系

```



## 11.LOG 算子检测边缘

用 MATLAB 程序实现如下：

```

I=imread('hands1.jpg');
subplot(2,2,1);
imshow(I);
title('原始图像');
I1=rgb2gray(I);
subplot(2,2,2);
imshow(I1);
title('灰度图像');
I2=edge(I1,'log');
subplot(2,2,3);

```

```
imshow(I2);  
title('log 算子分割结果');
```

原始图像



灰度图像



**log**算子分割结果



## 12.Canny 算子检测边缘

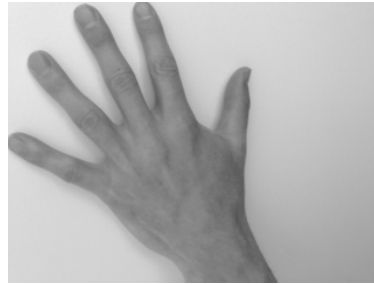
用 MATLAB 程序实现如下：

```
I=imread('hands1.jpg');  
subplot(2,2,1);  
imshow(I);  
title('原始图像')  
I1=rgb2gray(I);  
subplot(2,2,2);  
imshow(I1);  
title('灰度图像');  
I2=edge(I1,'canny');  
subplot(2,2,3);  
imshow(I2);  
title('canny 算子分割结果');
```

原始图像



灰度图像



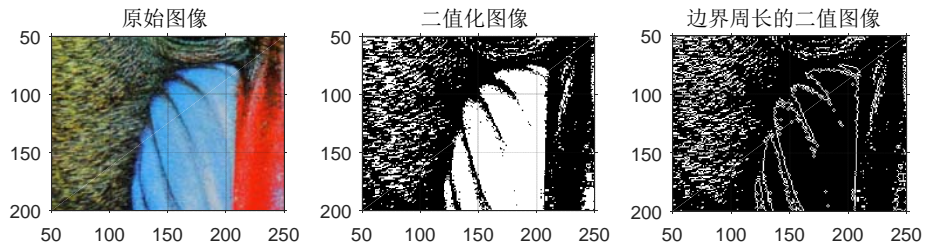
canny算子分割结果



### 13.形态学边界提取

利用 MATLAB 实现如下：

```
I=imread('baboon.bmp'); %载入图像
subplot(1,3,1),imshow(I);
title('原始图像');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系
I1=im2bw(I);
subplot(1,3,2),imshow(I1);
title('二值化图像');
axis([50,250,50,200]);
grid on; %显示网格线
axis on; %显示坐标系
I2=bwperim(I1); %获取区域的周长，边缘提取
subplot(1,3,3),imshow(I2);
title('边界周长的二值图像');
axis([50,250,50,200]);
grid on;
axis on;
```

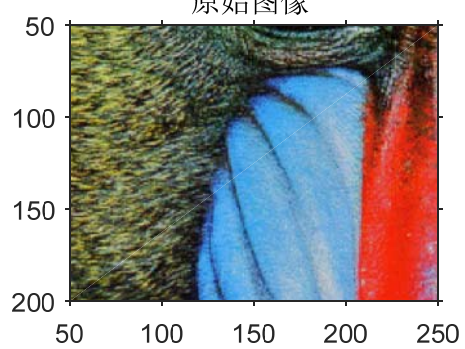


#### 14.形态学骨架提取

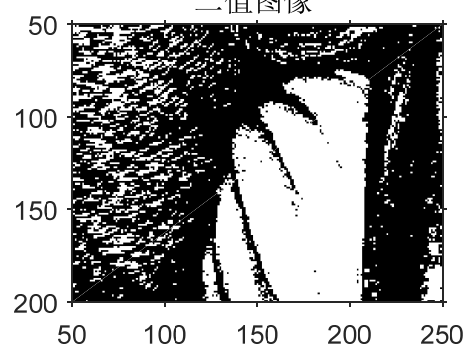
利用 MATLAB 实现如下：

```
I=imread('baboon.bmp');
subplot(2,2,1),imshow(I);
title('原始图像');
axis([50,250,50,200]);
axis on;
I1=im2bw(I);
subplot(2,2,2),imshow(I1);
title('二值图像');
axis([50,250,50,200]);
axis on;
I2=bwmorph(I1,'skel',1);
subplot(2,2,3),imshow(I2);
title('1 次骨架提取');
axis([50,250,50,200]);
axis on;
I3=bwmorph(I1,'skel',2);
subplot(2,2,4),imshow(I3);
title('2 次骨架提取');
axis([50,250,50,200]);
axis on;
```

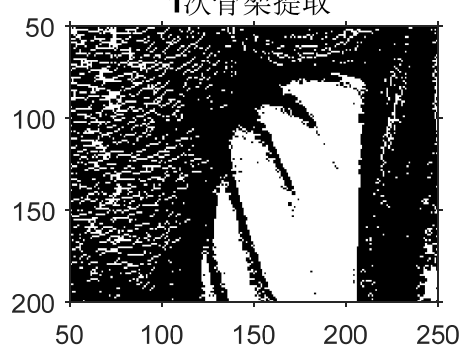
原始图像



二值图像



1次骨架提取



2次骨架提取

