

## Lab 2 - solution

### Step 1

```
ls
```

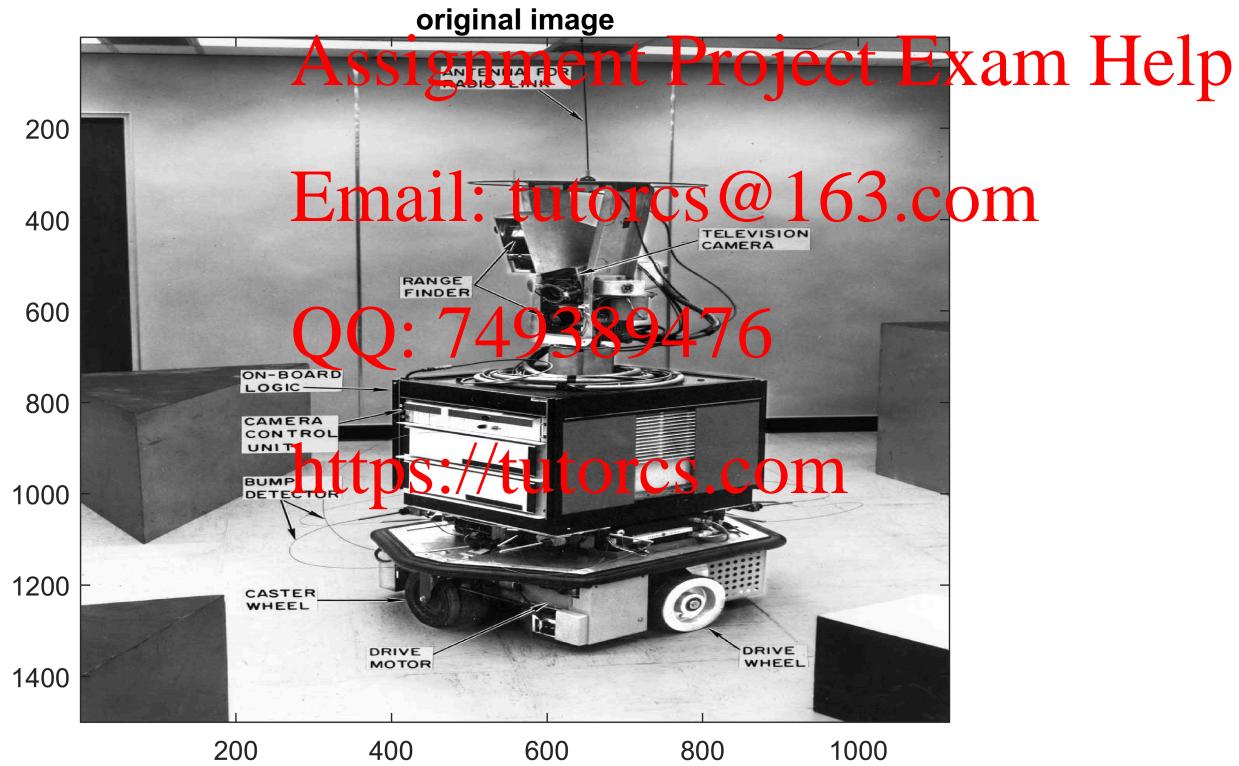
```
.
```

```
..  
N.m  
_MACOSX  
cluttera1.jpg  
face.gif  
filters.mat  
magnitude.m  
read_image.m  
roberts.mat  
shakey.1  
show_im  
sobel.ma  
solution
```



```
% Read and show image  
shakey = read_image('','shakey.150.gif');  
show_image(shakey)  
title("original image")
```

WeChat: cstutorcs



```
% Inspect filters  
gaussian_filter_5x5
```

```
gaussian_filter_5x5 = 5x5
0.0030 0.0133 0.0219 0.0133 0.0030
0.0133 0.0596 0.0983 0.0596 0.0133
0.0219 0.0983 0.1621 0.0983 0.0219
0.0133 0.0596 0.0983 0.0596 0.0133
0.0030 0.0133 0.0219 0.0133 0.0030
```

程序代写代做 CS 编程辅导

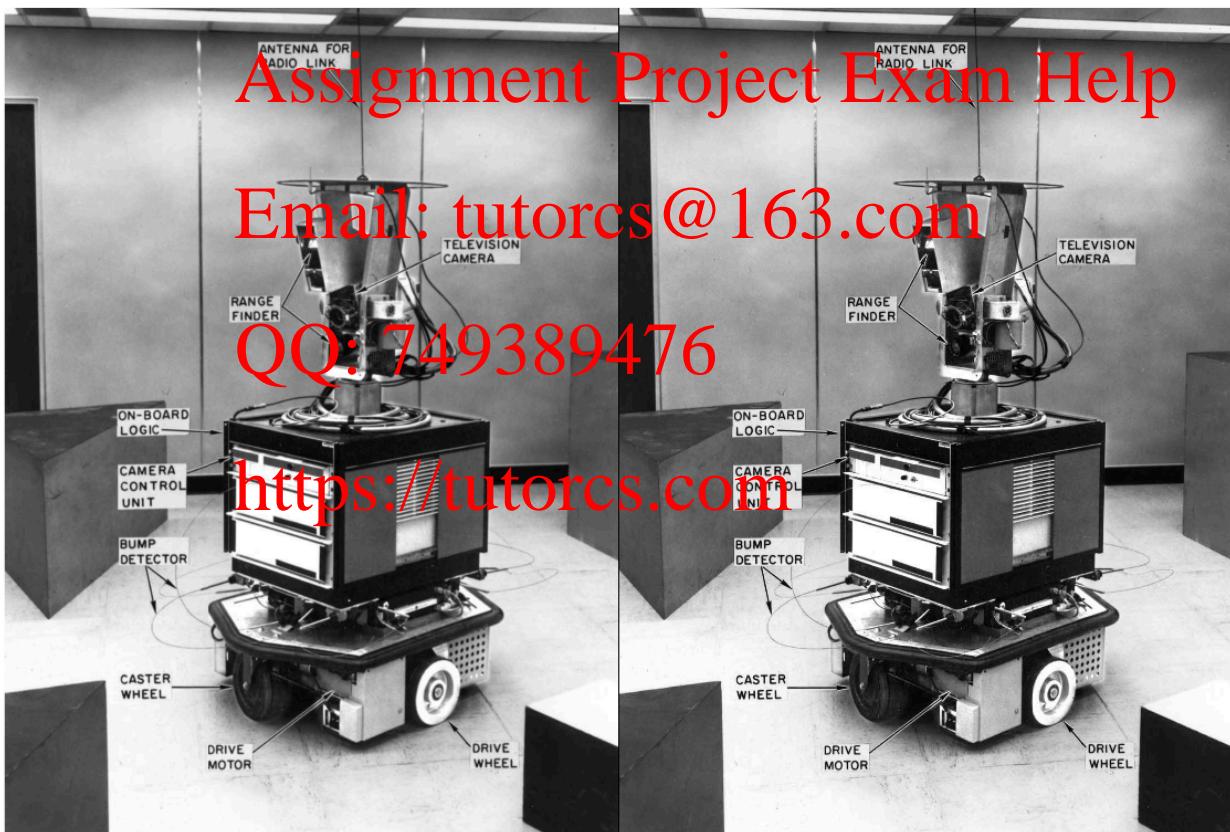
```
gaussian_filter_3x3
```

```
gaussian_filter_3x3 = 3x3
0.0113 0.0838 0.0
0.0838 0.6193 0.0
0.0113 0.0838 0.0
```



```
% Use convolve image
shakey_gaussaian_5x5 = conv2(shakey,gaussian_filter_5x5,'valid');
shakey_gaussaian_3x3 = conv2(shakey,gaussian_filter_3x3,'valid');
imshowpair(shakey_gaussaian_5x5,shakey_gaussaian_3x3,'montage')
```

WeChat: cstutorcs



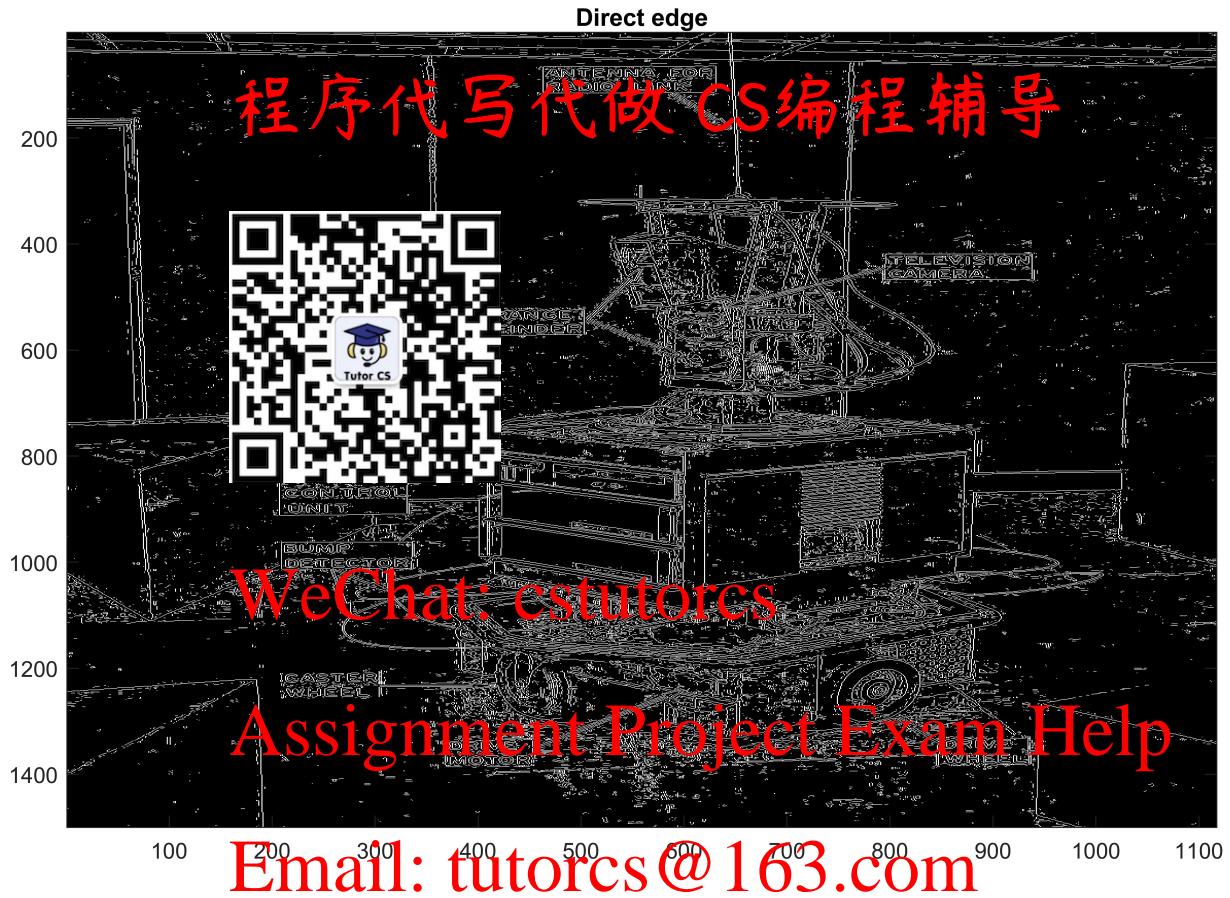
```
% Adding edge filter and thresholding
threshold = 5;
shakey_gaussaian_5x5_edge = edge(shakey_gaussaian_5x5,'Sobel',threshold);
shakey_gaussaian_3x3_edge = edge(shakey_gaussaian_3x3,'Sobel',threshold);
```

```
imshowpair(shakey_gaussain_5x5_edge, shakey_gaussain_3x3_edge, 'montage')
```



## Task 1

```
% Apply edge filter directly  
direct_edge = edge(shakey, 'obel', threshold),  
show_image(direct_edge)  
title("Direct edge")
```



Without smoothing from Gaussian filter, the image is noiser.

## Step 2

<https://tutorcs.com>

```
% Create discrete sample from Gaussian
% For a 9x9 mask
m = 0;
s = 1;
x = [-4:1:4];
pdf_9x9 = N(m,s,x);
gaussian_filter_9x9 = pdf_9x9'*pdf_9x9;
shakey_gaussain_9x9 = conv2(shakey,gaussian_filter_9x9,'valid');
% show_image(shakey_gaussain_9x9)
% title("Gaussian 9x9, s="+s)
```

```
% Increasing s
m = 0;
s = 10;
x = [-4:1:4];
pdf_9x9 = N(m,s,x);
gaussian_filter_9x9 = pdf_9x9'*pdf_9x9;
shakey_gaussain_9x9_s_10 = conv2(shakey,gaussian_filter_9x9,'valid');
```

```
% show_image(shakey_gaussain_9x9_s_10)
% title("Gaussian 9x9, s="+s)
```

imshowpair(shakey\_gaussain\_9x9, shakey\_gaussain\_9x9\_s\_10, 'montage')



<https://tutorcs.com>

Image become blurrier.

```
% Now apply Sobel
shakey_sobelX = conv2(shakey_gaussain_9x9,sobelX,'valid');
shakey_sobelY= conv2(shakey_gaussain_9x9,sobelY,'valid');
mag_sob = magnitude(shakey_sobelX, shakey_sobelY);

shakey_sobelX_s_10 = conv2(shakey_gaussain_9x9_s_10,sobelX,'valid');
shakey_sobelY_s_10 = conv2(shakey_gaussain_9x9_s_10,sobelY,'valid');
mag_sob_s_10 = magnitude(shakey_sobelX_s_10, shakey_sobelY_s_10);

imshowpair(mag_sob,mag_sob_s_10, 'montage')
```



Edges becomes blurred.

QQ: 749389476

## Task 2

```
imshowpair(shakey_gaussian_5x5,shakey_gaussian_3x3, 'montage')
```



Increasing Gaussian filter size led to a decrease in noise.

**QQ: 749389476**

```
imshowpair(shakey_gaussain_9x9,shakey_gaussain_9x9_s_10,'montage')
```

<https://tutorcs.com>



Email: tutorcs@163.com

QQ: 749389476

It also blurs the image. This is because  $s$  for a two-dimensional kernel is the radius in pixels containing 68% of the integrated magnitude of the coefficients. Increasing the standard deviation will increase the effective kernel size.

<https://tutorcs.com>

### Step 3

```
% Compare speed
% 2D
tic
large_filter = N(0,1,[-8:1:8]);
gaussian_filter = conv2(large_filter, large_filter');
image = conv2(shakey,gaussian_filter, 'valid');
T1 = toc
```

T1 = 0.1218

```
% 2 times 1D
tic
large_filter = N(0,1,[-8:1:8]);
shakey_1 = conv2(shakey, large_filter, 'valid');
shakeyfinal = conv2(shakey_1, large_filter', 'valid');
```

```
T2 = toc
```

```
T2 = 0.0541
```

Two large 1D in sequence is faster (50%). As mask sizes increase, the CPU time is larger.

```
% Checking two resulting image  
sum(image == shakeyfinal) // (1481*1101)*100
```

```
ans = 9.0544
```

Only 9% of pixels are same.

```
% Display image  
imshowpair(image,shakeyfinal)
```



Can't tell any visual differences.

## Step 4

```
% Laplacian filter  
image_laplacian = conv2(shakey,laplacian,'valid');
```

```
I_out = edge(image_laplacian, 'zerocross');
% show_image(I_out)
```

The resulting image is very noisy. This because Laplacian filter is the sum of two second order derivatives in x- and y- axis which amplifies noise easily.

```
% LoG refers to Gaussian filter first then apply Laplacian filter
image_laplacian = conv2(gaussian_filter('gaussian', 9, 10), laplacian, 'valid');
I_out_Log = edge(image_laplacian, 'zerocross');
```

```
% Comparison
imshowpair(I_out, I_out_Log)
```

WeChat: cstutorcs  
Assignment Project Exam Help  
Email: tutorcs@163.com  
QQ: 749389476  
<https://tutorcs.com>