第三次作业报告

夏熙 3017218170

2019 年 12 月 21 日

1 第一题

对LoG的数学形式进行数学推导（连续型）推导过程如图所示。
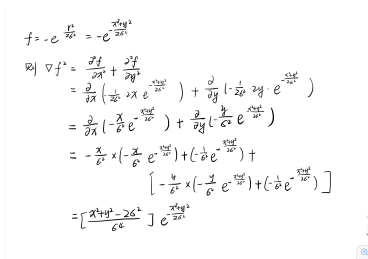

图 1: 推导过程

2 第二题

对于直线方程y=ax+b，生成一系列纵坐标符合高斯分布的点（提示，random出x，再算出y，再给y加入高斯噪声），再人工加入一系列的outlier，使用最小二乘法、RANSAC法、霍夫变换法拟合一条直线。

2.1 最小二乘法

理解最小二乘法：因为 y=ax+b，可令F=ax+b-y，那么对于模型上的点(注意是模型上的点，也就是理论值)，F=ax+b-y=0。但是对于实际值来说，

$$F = ax_i + b - y_i$$

一定不等于0。那么我们就要找到一对a和b，使得F尽可能接近于0。也就是说，"偏离量总和最小"这个概念，在数学上实际上就是要求F的方差最小。即

$$\Sigma F^2 \to 0$$

（F的平方和趋近于0），即

$$\Sigma(ax_i + b - y_i)^2 \to 0$$

。那么我们得到一个方程f(a,b)=

$$\Sigma(ax_i + b - y_i)^2$$

，我们要找到合适的a,b使得f(a,b)=

$$\Sigma(ax_i + b - y_i)^2$$

最小。

通过找极值点：

$$\delta(\Sigma(ax_i + b - y_i)^2)/\delta a = 0$$

，

$$\delta(\Sigma(ax_i + b - y_i)^2)/\delta b = 0$$

化简得到

$$a * \Sigma x_i^2 + b * \Sigma x_i = \Sigma(x_i * y_i)$$

，

$$a * \Sigma x_i + b * N = \Sigma y_i$$

解上面的二元方程，我们就可以得到唯一的一组a,b就是我们所需要的a和b

To list your MATLAB code:

```
1    function []=leastSquareMethod()
2    clear;
3    size = 200;
4    x=rand(1,size);
5    y=x;
6    g=imnoise(y,'gaussian',0.01,0.002);
```

```matlab
7       noisenum=20;
8       noise1=rand(1,noisenum)*(max(x)-min(x))+min(x);
9       noise2=rand(1,noisenum)*(max(x)-min(x))+min(x);
10
11      x=[x noise1];
12      y=[g noise2];
13      scatter(x,y);hold on;
14
15          x2 = sum(x*x');
16          x1 = sum(x);
17          xy = sum(x*y');
18          y1 = sum(y);
19      a=(length(x)*xy-x1*y1)/(length(x)*x2-x1*x1);
20      b=(y1-a*x1)/length(x);
21
22      y2=x*a+b;
23      plot(x,y2,'r');
24      title(['Stright line is:  y =  ',num2str(a),'x + '
            ,num2str(b)]);
```
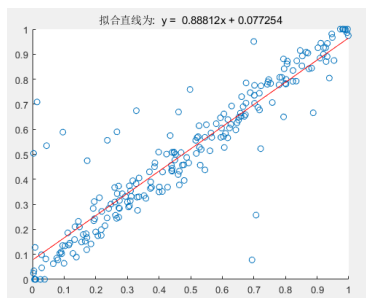
处理结果如图所示

图 2: 处理结果

2.2   RANSAC法

To list your MATLAB code:

3

```matlab
1    function []=RANSAC()
2    clear;
3    psize = 200;
4    x=rand(1,psize);
5    y=x;
6    g=imnoise(y,'gaussian',0.01,0.002);
7    noisenum=20;
8    noise1=rand(1,noisenum)*(max(x)-min(x))+min(x);
9    noise2=rand(1,noisenum)*(max(x)-min(x))+min(x);
10
11   x=[x noise1];
12   y=[g noise2];
13   scatter(x,y);hold on;
14   data = [x' y']';
15   number = psize+noisenum;
16   sigma=1;
17   pretotal=0;
18
19   for i=1:100
20       idx = randperm(number,2);
21       sample = data(:,idx);
22       x = sample(1, :);
23       y = sample(2, :);
24       k=(y(1)-y(2))/(x(1)-x(2));
25       b = y(1) - k*x(1);
26       line = [k -1 b];
27       mask=abs(line*[data; ones(1,size(data,2))]);
28       total=sum(mask<sigma);
29       if total>pretotal
30           pretotal=total;
31           bestline=line;
32       end
33   end
```

```matlab
34    mask=abs(bestline*[data; ones(1,size(data,2))])<
          sigma;
35    k=1;
36    for i=1:length(mask)
37        if mask(i)
38            inliers(1,k) = data(1,i);
39            k=k+1;
40        end
41    end
42
43    k = -bestline(1)/bestline(2);
44    b = -bestline(3)/bestline(2);
45    x = min(inliers(1,:)):0.1:max(inliers(1,:));
46    y = k*x + b;
47    plot(x,y,'r');
48    title(['Line is:  y =  ',num2str(k),'x + ',num2str
          (b)]);
```

处理结果如图所示
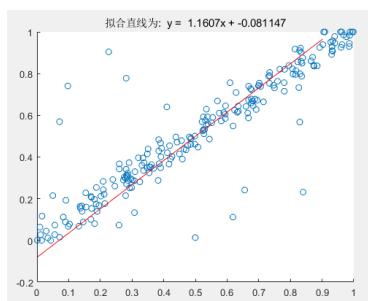
图 3: 处理结果

2.3   Hough变换法

霍夫变换，将(a,b)平面转化成

$$\rho\theta$$

5

平面，推导如图

$$y = ax + b$$
$$-ax + y = b$$
$$令 k = a^2 + 1$$
$$-\frac{a}{k} x + \frac{1}{k} y = \frac{b}{k}$$
$$\cos\theta\, x + \sin\theta\, y = \rho$$

图 4: (a,b)平面转化

To list your MATLAB code:

```matlab
function []=hough()
clear;
psize = 200;
x=rand(1,psize);
y=x;
g=imnoise(y,'gaussian',0.01,0.002);
noisenum=20;
noise1=rand(1,noisenum)*(max(x)-min(x))+min(x);
noise2=rand(1,noisenum)*(max(x)-min(x))+min(x);

x=[x noise1];
y=[g noise2];
scatter(x,y);
hold on;
data=[x;y];
number=psize+noisenum;
nma=5;
h=zeros(315,2*nma);
ti=1;
ma=80;
for theta=0:0.01:3.14
```

```
22      p=[cos(theta),sin(theta)];
23      d=p*data;
24       for i=1:number
25      h(ti,round(d(i)/10+nma))=h(ti,round(d(i)/10+nma))
            +1;
26       end
27       ti=ti+1;
28  end
29  [tx,p]=find(h>ma);
30  lines=size(tx);
31  r=(p-nma)*10;
32  tx=0.01*tx;
33  x=min(data(:)):0.05:max(data(:));
34      for i=1:40:lines
35          y = cot(tx(i))*x+r(i)/sin(tx(i));
36          plot(x,y,'r');
37      end
38  end
```
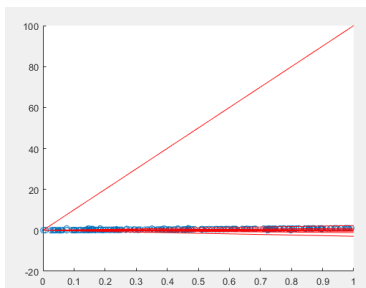
处理结果如图所示

图 5: 处理结果

3   第三题

找到一副实际的图像，使用一阶导数或二阶导数找出边缘点，使用上述几种方法，找到其中的直线这里用的是sobel算子（一阶导数）来确定边

7

缘点

3.1   最小二乘法

代码如下

```
1           function []=LSPro()
2           clear;
3           RGB = imread('snow.jpg');
4           I = double (rgb2gray(RGB));
5           esobel=edge(I,'sobel');
6           imshow(I);
7           imshow(esobel);
8           [imgH,imgW]=size(esobel);
9           p=1;
10          for i=1:imgH
11              for j=1:imgW
12                      if esobel(i,j) ~= 0
13                              x(p)=i;
14                              y(p)=j;
15                              p=p+1;
16                      end
17              end
18          end
19          hold on;
20          for i=20:20:400
21              x1=x(:,i-19:i);
22              y1=y(:,i-19:i);
23              a = x1*x1';
24              b = sum(x1);
25              c = x1*y1';
26              d = sum(y1);
27              k = (length(x1).*c-b*d)./(length(x1).*a-b*
                    b);
28              t = (a.*d-c.*b)/(a*length(x1)-b.*b);
```

```matlab
29              y1=x1*k+t;
30              plot(x1,y1,'r');
31         end
```

3.2 RANSAC法

代码如下

```matlab
1              function []=RANPro()
2              clear;
3             RGB = imread('snow.jpg');
4             I = double (rgb2gray(RGB));
5             esobel=edge(I,'sobel');
6             imshow(I);
7             imshow(esobel);
8             [imgH,imgW]=size(esobel);
9             p=1;
10            for i=1:imgH
11                for j=1:imgW
12                        if esobel(i,j) ~= 0
13                            x(p)=i;
14                            y(p)=j;
15                            p=p+1;
16                        end
17                end
18            end
19            hold on;
20            data = [x' y']';
21            number = size(data,2)
22            sigma=1;
23            pretotal=0;
24
25            for i=1:100
26                idx = randperm(number,2);
```

```matlab
27                    sample = data(:,idx)
28                    x = sample(1, :)
29                    y = sample(2, :);
30                    k=(y(1)-y(2))/(x(1)-x(2));
31                    b = y(1) - k*x(1);
32                    line = [k -1 b];
33                    mask=abs(line*[data; ones(1,size(data
                          ,2))]);
34                    total=sum(mask<sigma);
35                    if total >25
36                        pretotal=total;
37                        bestline=line;
38                        mask=abs(bestline*[data; ones(1,
                              size(data,2))])<sigma;
39                        k=1;
40                        for i=1:length(mask)
41                            if mask(i)
42                                inliers(1,k) = data(1,i);
43                                k=k+1;
44                            end
45                        end
46                        k = -bestline(1)/bestline(2);
47                        b = -bestline(3)/bestline(2);
48                        x = min(inliers(1,:)):0.1:max(
                              inliers(1,:));
49                        y = k*x + b;
50                        plot(x,y,'r');
51                    end
52                end
```
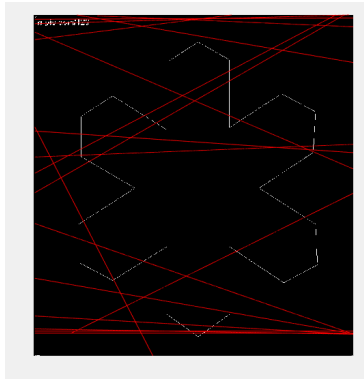
处理结果如图所示

3.3 Hough变换法

代码如下

图 6: 处理结果

```matlab
1   function []=HoughPro()
2   clear;
3   RGB = imread('snow.jpg');
4   I = double (rgb2gray(RGB));
5   esobel=edge(I,'sobel');
6   [H,T,R] = hough(esobel,'RhoResolution'
        , 0.5);
7   subplot(2,2,1);
8   imshow(I);
9   title('untrimmed');
10  subplot(2,2,2);
11  imshow(esobel);
12  title('Sobel');
13  subplot(2,2,3);
14  imshow(imadjust(mat2gray(H)),'XData',T
        ,'YData',R,'InitialMagnification','
        fit');
15  title('hough');
16  xlabel('\theta '),ylabel('\rho ');
17  axis on,axis normal,hold on;
18  colormap(hot);
```