

第 16 周电子版作业

胡恩 18343043

1 Q1

通用方法设计：

首先识别涂鸦处的 rgb 值，并获取涂鸦部分的位置参数，然后对涂鸦部分进行多次中值滤波（不采用均值滤波，虽然 psnr 值可能会变，但视觉效果有一定糊化），调整中值滤波半径以使得填充效果最好。经测试，滤波半径范围在 11:20 区间较好，迭代次数在 70 次足矣，过多迭代时间过长且视觉效果没有明显改善。

算法代码如下：

Listing 1: Code of Q1

```
1 clear;close all
2 %% 原图及坏图
3 org=imread('crayon_paint.png');
4 obj=imread('crayon_paint_thinscratch.png');
5 figure(1);subplot(1,2,1);imshow(org);subplot(1,2,2);imshow(obj);
6 %% 分析，坏点确实是rgb值为[0,0,0]的点
7 blakid=obj(:,:,1)==0 & obj(:,:,2)==0 & obj(:,:,3)==0;% rgb=[0,0,0]的点
8 obj_blak(:,:,1)=immultiply(blakid,255+obj(:,:,1));
9 obj_blak(:,:,2)=immultiply(blakid,255+obj(:,:,2));
10 obj_blak(:,:,3)=immultiply(blakid,255+obj(:,:,3));
11 figure(2)
12 imshow(uint8(255-obj_blak))
13 %% 选择性的中值滤波
14 res_psnr=zeros(10);
15 obj_filt=double(obj);obj_db=double(obj);
16 n=70;% 经调试，70次足以，迭代次数过多会运行太久且视觉效果没有明显提升
17 % r=11:20
18 % res_psnr =
```

```

19 %      36.2692    35.6366    35.9309    35.4596    35.8209    35.3355    ...
      35.6730    35.2106    35.5751    35.1366
20 for r=[11:20]
21     for ii=1:n
22         obj_filt(:,:,1)=medfilt2(obj_filt(:,:,1),[r,r]);
23         obj_filt(:,:,2)=medfilt2(obj_filt(:,:,2),[r,r]);
24         obj_filt(:,:,3)=medfilt2(obj_filt(:,:,3),[r,r]);
25         obj_filt(:,:,1)=obj_filt(:,:,1).*blakid+obj_db(:,:,1).*-blakid;
26         obj_filt(:,:,2)=obj_filt(:,:,2).*blakid+obj_db(:,:,2).*-blakid;
27         obj_filt(:,:,3)=obj_filt(:,:,3).*blakid+obj_db(:,:,3).*-blakid;
28     end
29     result=uint8(obj_filt);
30     res_psnr(r-10)=psnr(org,result);
31     % 将结果写入文件
32     fname=['result',int2str(r),'itr',int2str(n),'.png'];
33     imwrite(result,fname);
34 end

```

结果分析：

滤波半径 11 的 PSNR 值最高，但视觉效果并不好，可以明显的看到“303”的字样，仔细观察还能观察到“A”和“嘉哥的”的字样，图片如下：



图 1: Radius11Iter70

滤波半径为 15 的视觉效果最好，但仔细观察还是能看到“3”和“嘉”

的字样，图片如下：



图 2: Radius15Iter70

2 Q2

设计：

模拟球面上的基于元胞自动机的传染病模型（进程）。元胞自动机是一个模型，简单来讲：其中每个离散点的下一时刻的值由其附近的点决定。

此代码仍具有可拓展性，比如美化绘图，或调低某区域的感染概率或调高某区域的感染概率等。

代码及注释如下：

Listing 2: Code of Q2

```
1 %% 模拟基于元胞自动机的传染病模型，球面动态绘图
2 clear;clc;close all
3 %% 讨论范围
4 % 定为一个300*300的二维空间，可调整
5 n = 300; area=zeros(n);% 空间矩阵area
6 %% 模型notation及初始状态
```

输出的 gif 请参见附件

3 Q3(1) 复杂水印去除问题

方法设计：

对于一个灰度图像，一般最亮为底色白色，最暗为文字信息黑色，而中间（能区别于白色底色且不遮挡黑色文字）的即为水印，故此算法识别出底色和文字信息后设置硬阈值，直接滤去水印。

代码如下：

Listing 3: Code of Q3(1)

```
1 clear;close all
2 fname='Book_watermarked.png';
3 % fname='excel_elite.jpeg'; % another image
4 org=imread(fname);
5 % org=rgb2gray(org); % for a colorful image
6 brightest=max(org,[],'all');
7 darkest=min(org,[],'all');
8 % upper and lower fence
9 bright_fence=brightest-20;
10 dark_fence=darkest+80;
11
12 % filt index, identify the watermark
13 filtid=((org>dark_fence) & (org<bright_fence));
14
15 filt=org;
16 filt(filtid)=brightest; % set the watermark to white
17 res=uint8(filt);
18 figure(1)
19 imshow(org);
20 figure(2)
21 imshow(res);
```

结果如下：

§9.6 最大值原理

在前两节中我们看到了大量的连续函数, 虽然肯定并非所有的函数都是连续的. 我们现在证明连续函数具有许多其他的有用的性质, 特别是当它们的定义域是闭区间的时候是这样. 正是在这里, 我们将开始把 Heine-Borel 定理 (定理 9.1.24) 的全部功效予以展示.

定义 9.6.1 设 X 是 \mathbb{R} 的子集合, 并设 $f: X \rightarrow \mathbb{R}$ 是函数. 如果存在实数 M , 使得对于一切 $x \in X$ 都有 $f(x) \leq M$, 我们就说 f 是有上界的; 如果存在实数 M , 使得对于一切 $x \in X$ 都有 $f(x) \geq -M$, 我们就说 f 是有下界的; 如果存在实数 M , 使得对于一切 $x \in X$ 都有 $|f(x)| \leq M$, 我们就说 f 是有界的,

注 9.6.2 一个函数是有界的, 当且仅当它既是有上界的也是有下界的 (为什么? 注意 “当且仅当” 的一部分比另一部分稍许微妙). 同时, 函数 $f: X \rightarrow \mathbb{R}$ 是有界的, 当且仅当它的象 $f(X)$ 是依定义 9.1.22 的意义有界的 (为什么?).

并非一切连续函数都是有界的. 例如函数 $f(x) := x$ 在定义域 \mathbb{R} 上是连续的, 然而它是无界的 (为什么?), 尽管它在某些更小的区域上是有界的, 例如在 $[1, 2]$ 上它

图 3: OriginalBookWatermarked

§9.6 最大值原理

在前两节中我们看到了大量的连续函数, 虽然肯定并非所有的函数都是连续的. 我们现在证明连续函数具有许多其他的有用的性质, 特别是当它们的定义域是闭区间的时候是这样. 正是在这里, 我们将开始把 Heine-Borel 定理 (定理 9.1.24) 的全部功效予以展示.

定义 9.6.1 设 X 是 \mathbb{R} 的子集合, 并设 $f: X \rightarrow \mathbb{R}$ 是函数. 如果存在实数 M , 使得对于一切 $x \in X$ 都有 $f(x) \leq M$, 我们就说 f 是有上界的; 如果存在实数 M , 使得对于一切 $x \in X$ 都有 $f(x) \geq -M$, 我们就说 f 是有下界的; 如果存在实数 M , 使得对于一切 $x \in X$ 都有 $|f(x)| \leq M$, 我们就说 f 是有界的,

注 9.6.2 一个函数是有界的, 当且仅当它既是有上界的也是有下界的 (为什么? 注意 “当且仅当” 的一部分比另一部分稍许微妙). 同时, 函数 $f: X \rightarrow \mathbb{R}$ 是有界的, 当且仅当它的象 $f(X)$ 是依定义 9.1.22 的意义有界的 (为什么?).

并非一切连续函数都是有界的. 例如函数 $f(x) := x$ 在定义域 \mathbb{R} 上是连续的, 然而它是无界的 (为什么?), 尽管它在某些更小的区域上是有界的, 例如在 $[1, 2]$ 上它

图 4: FiltedBookWatermarked

对题目给定图像处理结果较好，而对任意网上找到的图像效果不佳。

公司名称		日期:	日期
在此处键入公司标语		发票 #:	100
		客户 ID:	ABC12345
受票方	发货目的地		
姓名	姓名		
公司名称	公司名称		
街道地址	街道地址		
街道地址, 邮政编码	街道地址, 邮政编码		
电话	电话		
业务		付款方式	
业务或项目		收到收据后付款	
说明		金额	
对过期余额收取 1.5% 财务费用			
[日期] ¥ 1000 的发票 #100			

图 5: OriginalExcelWatermarked

公司名称		日期:	日期
在此处输入公司标识		岁数:	100
		客户 ID:	ABC12345
发货方		发货目的地	
姓名	姓名	姓名	姓名
公司名称	公司名称	公司名称	公司名称
街道地址	街道地址	街道地址	街道地址
街道地址: 邮政编码	街道地址: 邮政编码	街道地址: 邮政编码	街道地址: 邮政编码
电话	电话	电话	电话
业务			付款方式
业务名称	业务描述		业务日期
说明			金额
对逾期余额收取 1.5% 的服务费			
[日期] * 1000 的发票 = 100			

图 6: FiltedExcelWatermarked

4 Q3(3) 半盲反卷积问题

算法设计:

此算法设计参考了 [1][2][3]。

算法主要思想是先通过分析模糊图像的频域图，从而估计移动模糊核的参数幅度 Length 和角度 Theta。再利用估计的移动模糊核，对于 circular 边界条件，直接应用反卷积函数 deconvreg 恢复图像；对于 replicate 边界条件，应用解线性反问题的 gmres 函数恢复图像。

4.1 核估计

对于一个受到移动模糊的图像，其傅里叶变换的频域图像会呈现与移动方向垂直的光带。如图：

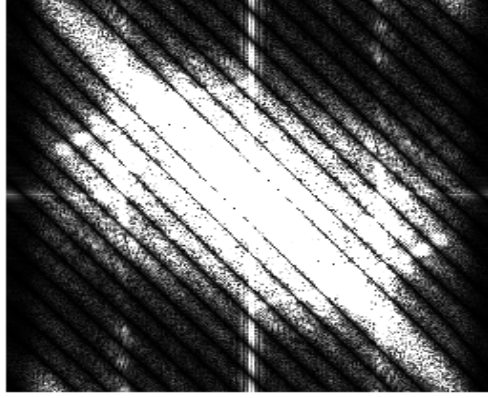


图 7: FourierTransformOfMotionBlurCircularWith20Length40Degree

4.1.1 角度估计

我们利用移动模糊频域图像的性质，即光带的性质来估计角度 Theta。考虑一个与频域上 η 轴负方向夹角为 θ ，与原点 origin 距离为 ρ 的直线，我们考虑某图像 f 沿该直线的积分：

$$R(f, \rho, \theta) = \int_{-d}^d f(\rho \cos \theta - s \sin \theta, \rho \sin \theta + s \cos \theta) ds, d = \frac{N}{\sqrt{2}}$$

其中 N 为待 deblur 方形图像的边长（像素个数），这种方法常被称为 Radon Transform Method，以下简称 RT。

将 RT 应用在之前提到的频域图像上，则积分值代表沿某条直线的亮度之和。于是，一个对角度 Theta 的直观的估计是

$$\hat{\theta} = \arg \max_{\theta} \sum_{largest 5 \rho} R(\log |G(\xi, \eta)|, \rho, \theta),$$

其中 G 为模糊图的频域图像，而由于 G 在中心有极端值（如图），故取 \log 。在文献 [3] 中，使用了更加直观の利用方差估计，

$$\hat{\theta} = \arg \max_{\theta} Var\{R(\log |G(\xi, \eta)|, \rho, \theta)\}$$

，但实验效果不佳。

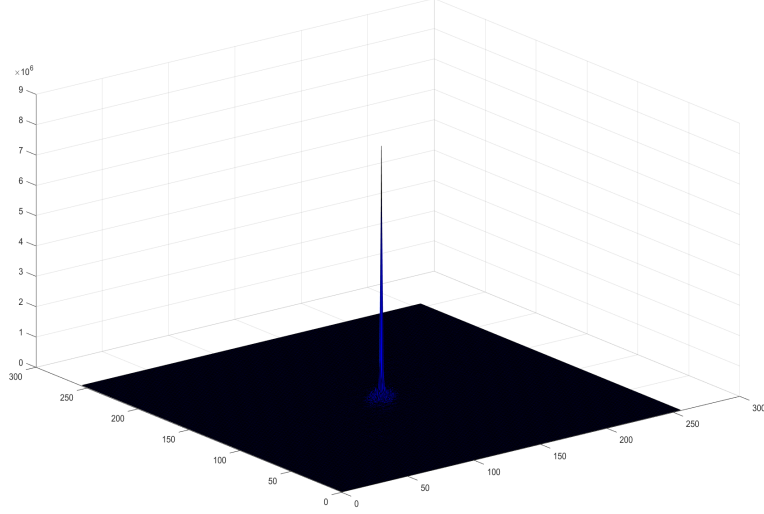


图 8: SurfOfAbsoluteG

4.1.2 幅度估计

对于移动模糊幅度 Length 的估计需要利用角度估计的结果 $\hat{\theta}$ 。

当我们成功估计了角度 $\hat{\theta}$ 后，将其看作真实的移动模糊角度。顺时针旋转频域图 $\hat{\theta}$ ，使得光带与 η 轴平行，此时，沿 ξ 轴，能量值（或模）大小服从函数

$$H(u) = \frac{\sin(\frac{Lu\pi}{N})}{L \sin(\frac{u\pi}{N})}$$

，其中 L 为移动模糊幅度 Length， u 为 ξ 轴坐标， N 为方形图像边长（像素个数）。

显然，这个函数具有周期性零点，根据频域图像我们可以获得零点 $\frac{Nk}{L}, k = 1, 2, \dots$ ，从而得到每个零点的间隔 $\frac{N}{L}$ ，从而利用 N 估计出 L 。

然而，在数值计算上，零点往往以一个极小的不为 0 的极小值出现，而非作为 0 出现，故我们转而寻找极小值点而非零点。另外，为了利用全图信息来找极小值点，我们对对应的每个 u 值，按 η 轴方向积分对应的 $\log |G|$ ，所找极小值点等价于找积分的极小值点，绘制出积分的图像，往往如图形式：

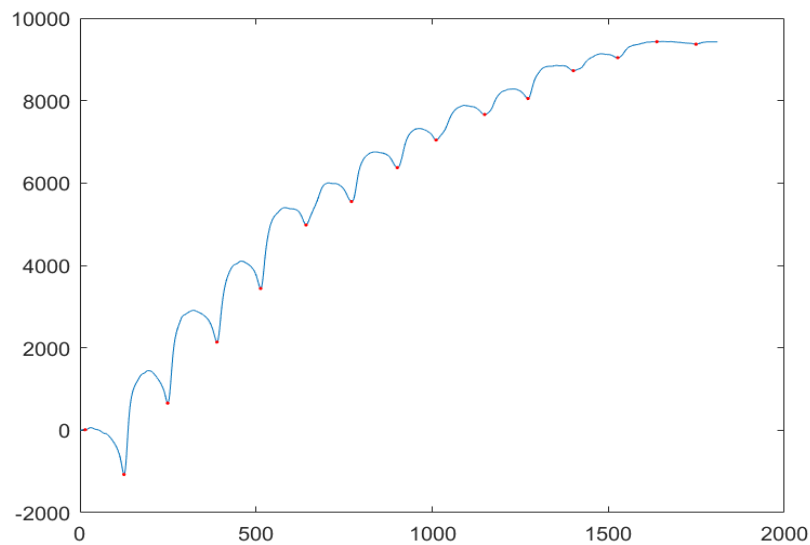


图 9: IntegralOfLogAbsG

利用找到的极小值点的间隔，取前 5 个间隔的中位数，除 N 即得移动模糊幅度 Length 的估计。

核估计的 MATLAB 代码如下：

Listing 4: ThetaLengthEstimate

```

1  function [est_theta,est_len]=ThetaLengthEstimate(g,ifplotfft,ifSrcBestS)
2
3  if nargin==1
4      ifplotfft=false;
5      ifSrcBestS=false;
6  elseif nargin==2
7      ifSrcBestS=false;
8  end
9
10 g=double(g);
11
12 [N,-]=size(g);% get the size of g. g is square by default.
13 G=fft2(g);
14 G=fftshift(G);% fftshift is needed, or this function will return 0
15 lnG=log(abs(G));

```

```

16 % create zeros entries around g in order to integral numerically for ...
    convenience
17 lnG_ext=zeros(2*N);lnG_ext(N/2+1:3/2*N,N/2+1:3/2*N)=lnG;% means extend ...
    of lnG
18
19 d=N/sqrt(2); % the farthest point from the origin (0,0)
20 drho=0.1; % ! set the accuracy of separation of lines
21 rho=(0:drho:d)'; % set the distance of lines from the origin (0,0)
22 t=180; % !! set accuracy of theta, t=36  $\supset$  180/36=5 degree per interval
23 theta=linspace(0,pi/t*(t-1),t);
24
25 %% Calculate matrix R=R(lnG,rho,theta), each entry correspond ...
    to(rho,theta)
26 lrho=length(rho);
27 S=zeros(lrho,t); % integral sum
28 ds=1;
29 % integrate step
30 for s=d:ds:d
31     x=rho.*cos(theta)-s.*sin(theta);
32     y=rho.*sin(theta)+s.*cos(theta);
33     colid=ceil(x/N);
34     rowid=ceil(N-y);
35     lin_id=sub2ind(size(lnG_ext),rowid,colid);
36     S=S+lnG_ext(lin_id);
37 end
38 clear x y colid rowid lin_id
39 R=S.*ds; % the result of integral
40
41 %% Use R to estimate theta
42 %% Method of find the largest variance
43 % VarRrho=var(R);
44 % [~,est_the_id]=max(VarRrho);
45 % est_theta=pi/t*(est_the_id-1);
46
47 %% Method of find the largest range, proposed by myself
48 % range=max(R)-min(R);
49 % [~,est_the_id]=max(range);
50 % est_theta=pi/t*(est_the_id-1);
51
52 %% Method of sum the largest 5 entries of each coloum,i.e. each theta
53 % refer to Felix Krahmer
54 Rsort=sort(R,'descend');
55 v=sum(Rsort(1:5,:));
56
57 % The estimate of angle theta is which has the max v
58 [~,est_the_id]=max(v);

```

```

59  est_theta=pi/t*(est_the_id-1);
60  % Given length of 20 motion, this estimate has a accuracy of 10:5:165
61  % and 0 of course , given circular boundary condition
62
63  %% Estimate Length
64  % This algorithm refers to the Joao P. A. Oliveira. Although there are ...
    some
65  % mistakes in the algorithm in the paper, I correct and improve it. The
66  % notations are same to the paper.
67
68  % using the estimated theta , integrate for different distance from ...
    origin
69  PI=R(:,est_the_id);
70  % standardized steps for the integral
71  Delta=diff(PI);
72  Delta_star=Delta;
73  p=3;
74  Delta_star(Delta>0)=p.*Delta(Delta>0);
75  S=cumsum(Delta_star);
76
77  % smooth to find desired minimal points
78  for ii=1:10
79      S=smooth(S);
80  end
81  minD=10/drho; % set the minimum of distance of closest minimal points
82  [~,id_peaks]=findpeaks(-S,'MinPeakDistance',minD); % find minimal points
83  % plot S
84  % figure;plot(S);hold ...
    on;plot(id_peaks,S(id_peaks),'r.','markersize',5);hold off;
85  id_peaks_dist=diff(id_peaks); % calculate distance , or called diff
86  d=median(id_peaks_dist(1:5)); % select the median of the first 5 ...
    distance
87  d=d*drho; % don't forget to multiply sample distance to obtain the ...
    true distance
88  est_len=floor(N/d); % estimate length
89  % est_len=round(N/d)-1;
90
91  if ifplotfft
92      figure
93      imshow(uint8(abs(G)))
94  end
95
96  if ifSrcBestS % Search Best S curve
97      est_the_id=est_the_id-5:est_the_id+5;
98      est_len=zeros(1,length(est_the_id));
99      for kk=1:length(est_the_id)

```

```

100     PI=R(:,est_the_id(kk));
101
102     Delta=diff(PI);
103     Delta_star=Delta;
104     p=3;
105     Delta_star(Delta>0)=p.*Delta(Delta>0);
106     S=cumsum(Delta_star);
107
108     for ii=1:10
109         S=smooth(S);
110     end
111     minD=10/drho; % set the minimum of distance of closest minimal ...
                  % points
112     [~,id_peaks]=findpeaks(-S, 'MinPeakDistance',minD); % find ...
                  % minimal points
113     figure;plot(S);hold ...
                  on;plot(id_peaks,S(id_peaks),'r.','markersize',5);hold off;
114     id_peaks_dist=diff(id_peaks); % calculate distance, or called ...
                  % diff
115     d=median(id_peaks_dist(1:5)); % select the median of the first ...
                  % 5 distance
116     d=d*drho; % don't forget to multiply sample distance to obtain ...
                  % the true distance
117     est_len(kk)=floor(N/d); % estimate length
118 end
119 est_theta=pi/t*(est_the_id-1);
120 end
121 end

```

4.2 反卷积

4.2.1 Circular 边界条件

Circular 边界条件是很理想的边界条件，上述核估计算法估计准确率非常高，对移动模糊角度 0 和 10:5:165 角度均能准确估计，对 0:1:179 角度估计误差在 4 度以内，对移动模糊幅度估计误差在 3 像素以内。

反卷积只需使用 deconvreg 函数即可，下图为复原 Circular 边界移动模糊幅度 20 角度 40 的结果，PSNR 值为 33



图 10: CircularBondaryMotionBluredBarbara



图 11: DebluredResult

4.2.2 Symmetric 边界条件

可以通过上下左右拼接，使其成为 Circular 边界条件，同样可以获得很好结果，略。

4.2.3 Replicate 边界条件及 Q3(3)

Q3(3) 是 Replicate 边界条件的 Motion Blur，这是因为在使用 deconvreg 函数去模糊时出现了 Replicate 边界条件特有的魔鬼图像，以下均以 Q3(3) 为例。

Replicate 边界是更难以处理的边界条件，其模糊图的频域图在水平和垂直方向中间光带很宽。如图



图 12: FourierTransformOfReplicateBoundaryBlur

这导致对角度 Theta 的估计有较大偏差，而不准确的角度估计会导致前述积分方向错误，从而导致直观上，前述积分值曲线规律性很差，找到的极小值点（零点）间隔实质上是大于真实方向上的极小值点间隔的，即估计的移动模糊核的幅度偏小。下图为给定的模糊图 Barbara 的错误估计角度（43 度）直线上的积分值曲线：

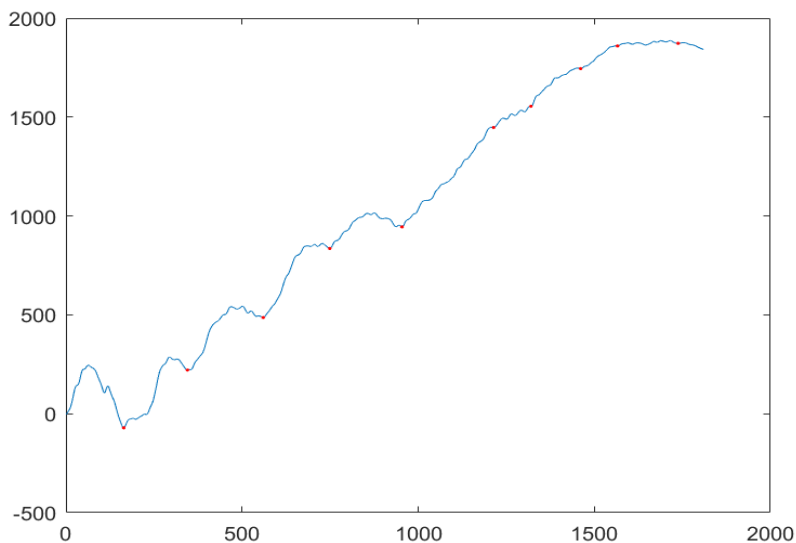


图 13: WrongIntegralOfLogAbsG

故我们绘出 43 度附近度数（即 43-5:1:43+5）的 S 曲线图，只需在控制台用这样一行语句调用函数 `ThetaLengthEstimate()`：

```
[est_theta,est_len]=ThetaLengthEstimate(g,false,true),
```

从中找出最符合具有周期性极小值这一特性的 S 曲线图，并找出对应度数为 39、40、41。

控制台输出结果为

Listing 5: Outcomes

1	est_theta =													
2	0.6632	0.6807	0.6981	0.7156	0.7330	0.7505	0.7679	...						
		0.7854	0.8029	0.8203	0.8378									
3	est_len =													
4	17	16	15	16	22	12	19	20	19	22	18			

下三图为 39、40、41 度的 S 曲线图

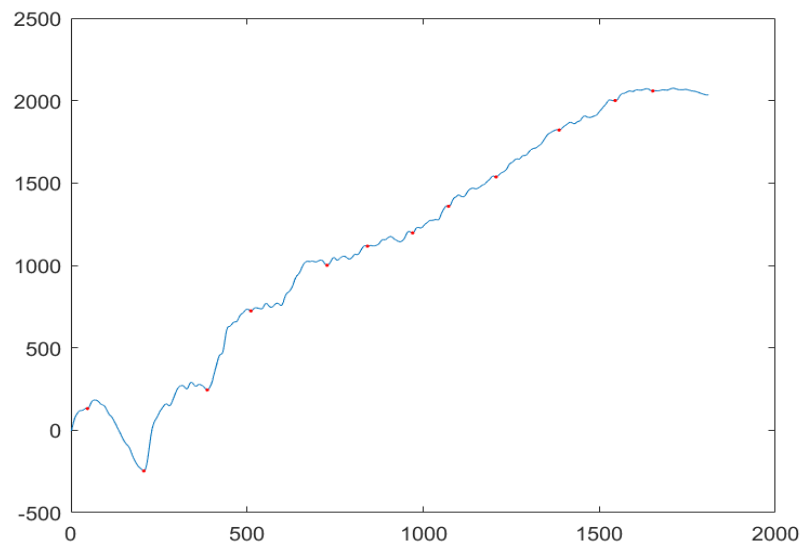


图 14: CurveSfor39degree

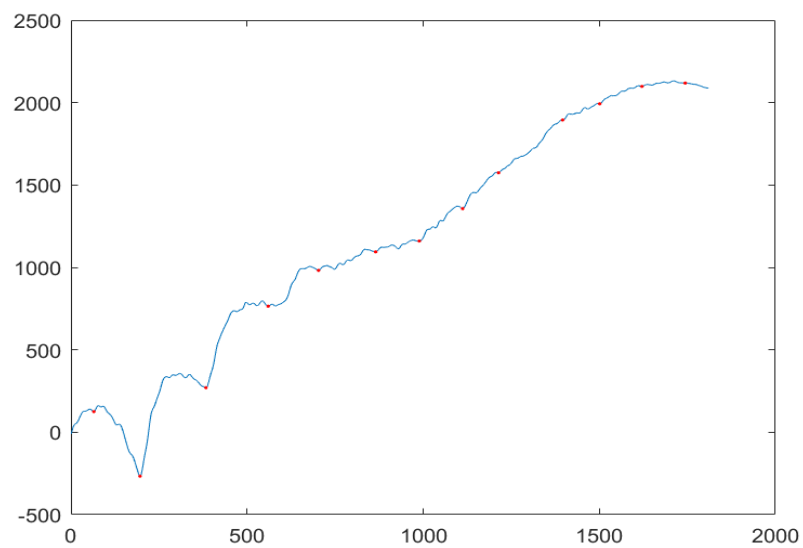


图 15: CurveSfor40degree

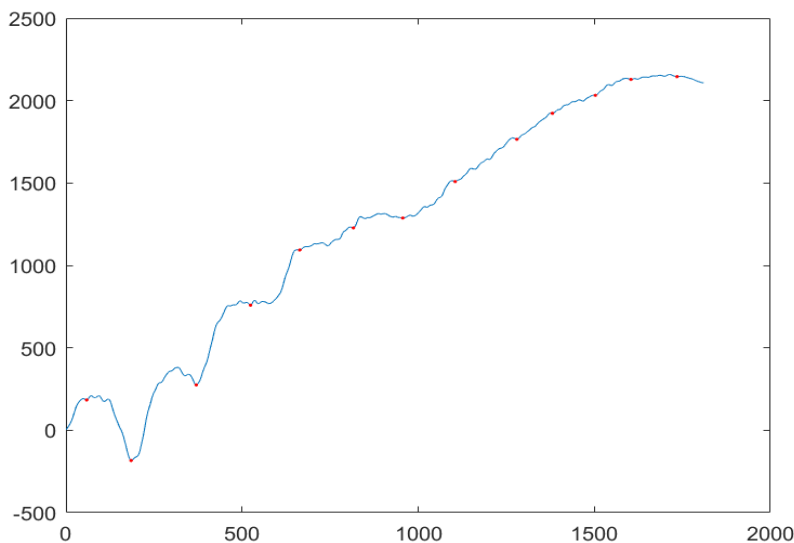


图 16: CurveSfor41degree

取角度估计为 40，并根据对应图像，取第 2、3 个标红点的明显极小值点间隔计算对应的幅度估计为 14

接下来，由于边界条件为 Replicate，将模糊图像反卷积问题看作解线性反问题，应用 MATLAB 自带函数 gmres() 求解。然而，gmres 对于移动 (Motion) 模糊的迭代求解可能由于机器误差不收敛，需通过调试得到最佳迭代次数。调试并观察反卷积结果得到最佳迭代次数为 17 次。

线性反问题法代码如下：

Listing 6: LinearInverse

```

1 function debl_r_f=lin_inv(g, kernel)
2 [m,n]=size(g);
3 g=double(g(:));
4 n_gmres=17; % best iteration times
5 debl_r_f=gmres(@(f) blurr(f,m,n, kernel),g,[],[],n_gmres);
6 debl_r_f=uint8(reshape(debl_r_f,m,n));
7 end
8
9 function g=blurr(f,m,n, kernel)
10 f=reshape(f,m,n);

```

```

11 % kernel=kernel/sum(kernel,'all');
12 g=imfilter(f,kernel,'replicate');
13 g=g(:);
14 end

```

解 Q3(3) 代码如下:

Listing 7: Q3(3)

```

1 clear;
2 close all
3 %% read object
4 g=imread('barbara_motionblurred.png');
5
6 %% test code block
7 % f=imread('barbara.png');
8 % f=double(f);
9 % kernel=fspecial('motion',20,40);
10 % g=imfilter(f,kernel,'replicate');
11
12 %% estimate motion kernel
13 % [est_theta,est_len]=ThetaLengthEstimate(g,false); % Default estimate
14 % [est_theta,est_len]=ThetaLengthEstimate(g,false,true); % Search best S
15 est_theta=40/180*pi; est_len=14; % Estimated by the curve of S for Q3(3)
16
17 %% print estimated result
18 disp('estimated theta is')
19 disp(est_theta);
20 disp('it equals to(degree):')
21 disp(est_theta/pi*180)
22 disp('estimated length is')
23 disp(est_len);
24 %% deblur
25 kernel=fspecial('motion',est_len,est_theta/pi*180);
26 f_deblur=lin_inv(g,kernel); % For replicate boundary condition
27 % f_deblur=deconvreg(g,kernel); % For circular boundary condition. One ...
    can deal
28 % with symmetric boundary condition making use of the method for ...
    circular.
29 %% psnr compute
30 f=imread('barbara.png');
31 psnr_deblur=psnr(uint8(f_deblur),uint8(f));
32 disp('Deblured psnr is')
33 disp(psnr_deblur);
34 g=uint8(g);
35 psnr_g=psnr(uint8(g),uint8(f))

```

```
36
37 %% imshow
38 figure
39 imshow(uint8(f_deblr))
40 figure
41 imshow(uint8(g))
42 figure
43 imshow(uint8(f))
```

控制台输出结果如下：

Listing 8: Outcomes

```
1 estimated theta is
2     0.6981
3 it equals to(degree):
4     40
5 estimated length is
6     14
7 gmres 在 迭代 17 停止，而没有收敛到所需容差 1e-06，
8 这是因为已达到最大迭代数。
9 迭代返回的（数目 17）的相对残差为 0.0041。
10 Deblured psnr is
11     26.5963
12 psnr_g =
13     22.8491
```

图像去模糊结果



图 17: OriginalBarbara



图 18: Q33MotionReplicateBlured



图 19: DeblurredLinearInverseMethod

我尽力了，感觉估计应该还是准确的，只是缺乏好的对 Replicate 边界条件的 Motion Blur 的反卷积算法，否则恢复效果应该会更好。

参考文献

- [1] Felix Krahmer, Youzuo Lin, Bonnie McAdoo, Katharine Ott, Jiakou Wang, David Widemann, and Brendt Wohlberg. Blind image deconvolution: Motion blur estimation. 2006.
- [2] Mohsen Ebrahimi Moghaddam and Mansour Jamzad. Motion blur identification in noisy images using fuzzy sets. In *Proceedings of the Fifth IEEE International Symposium on Signal Processing and Information Technology, 2005.*, pages 862–866. IEEE, 2005.
- [3] João P Oliveira, Mario AT Figueiredo, and José M Bioucas-Dias. Blind estimation of motion blur parameters for image deconvolution. In *Iberian Conference on Pattern Recognition and Image Analysis*, pages 604–611. Springer, 2007.