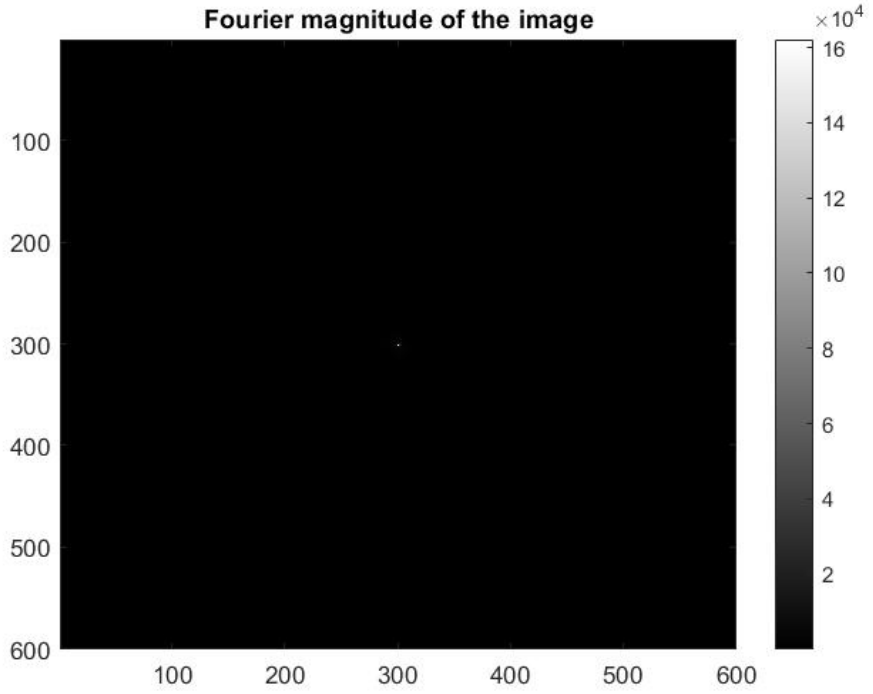
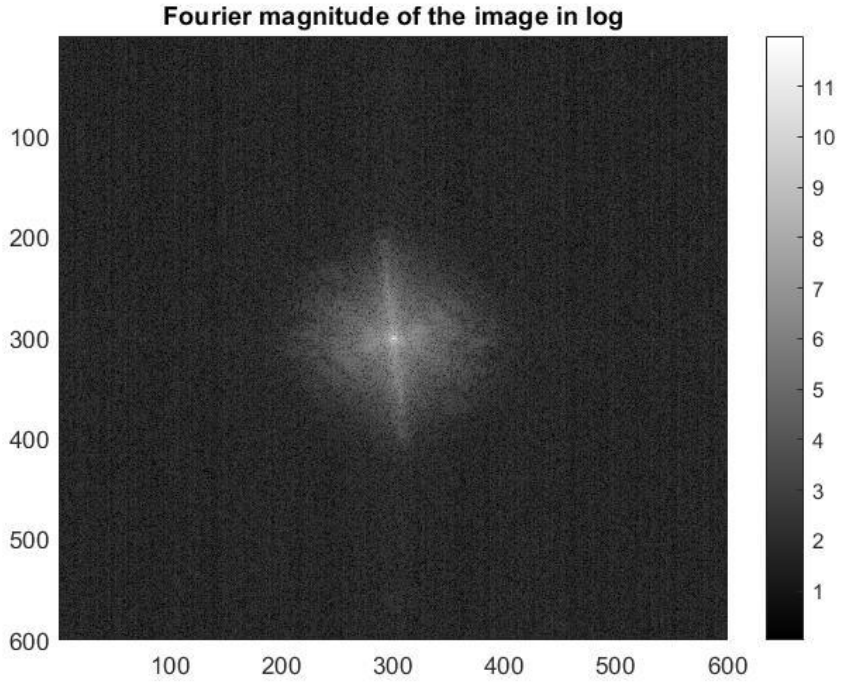
Project #3

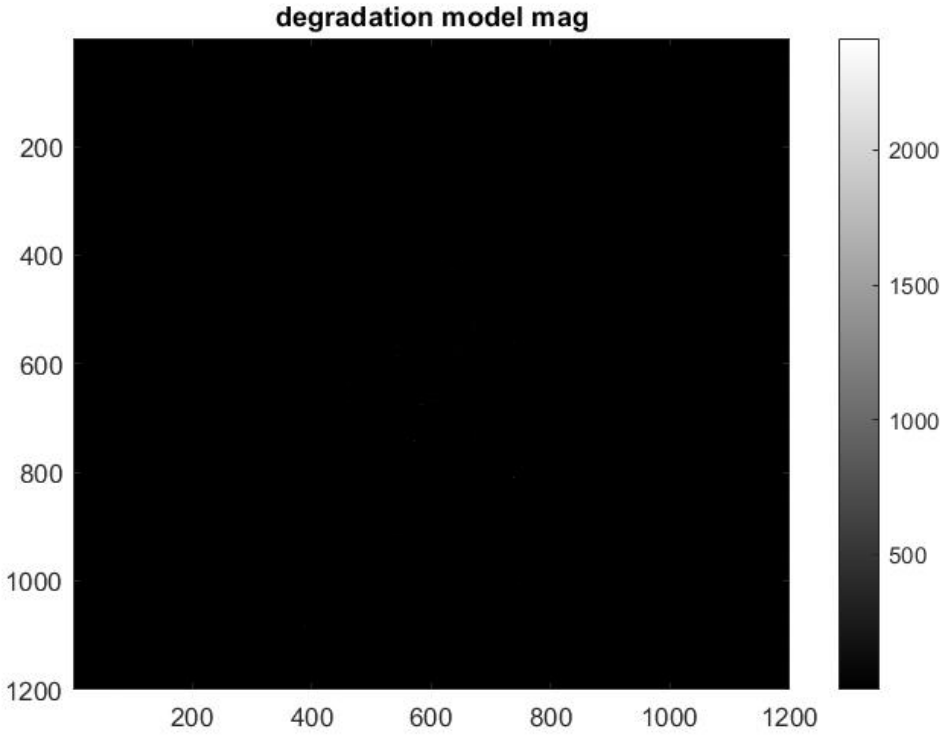
309605004 歐思辰

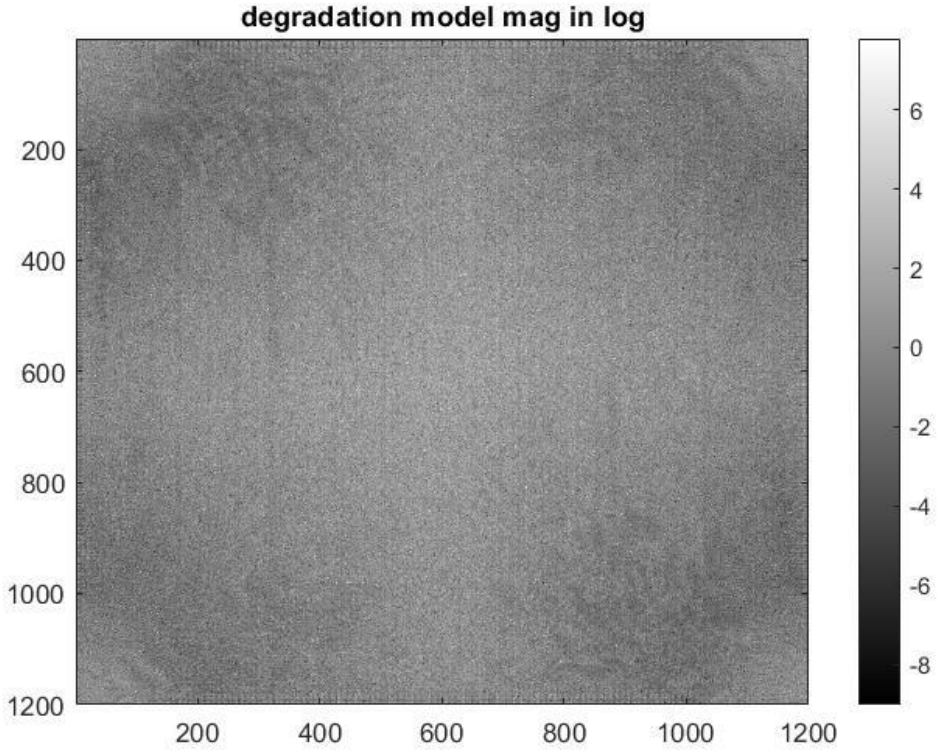
- Figure of the Fourier magnitude spectrum of the degraded image Bird 2 degraded



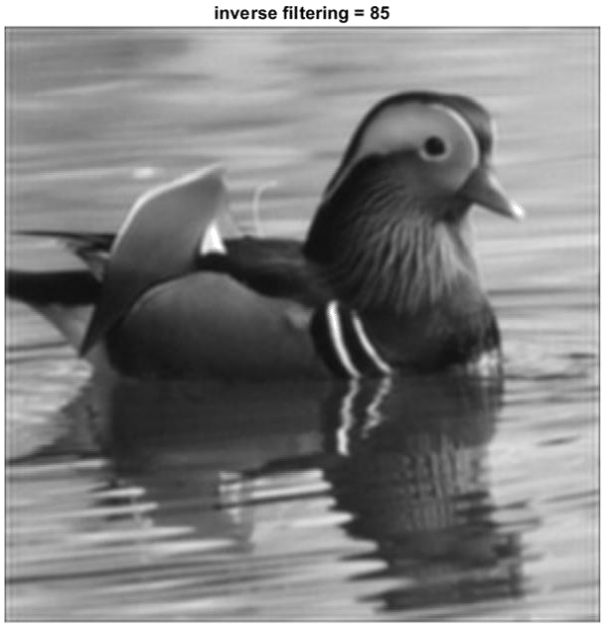


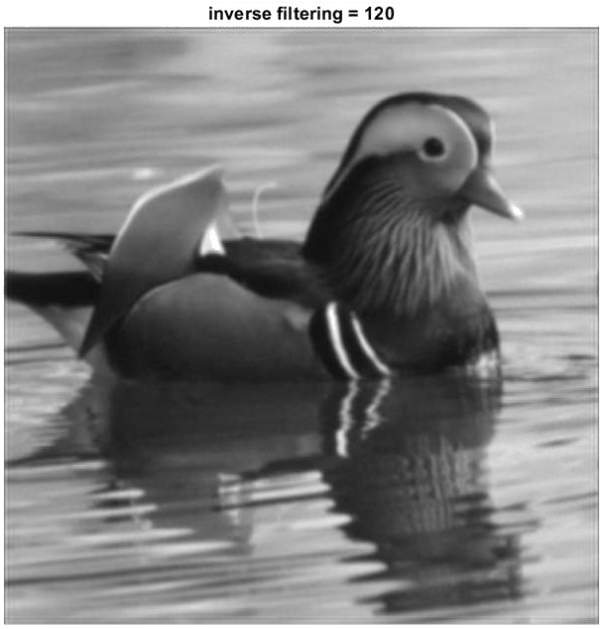
- Figure of the Fourier magnitude (frequency response) of degradation model H



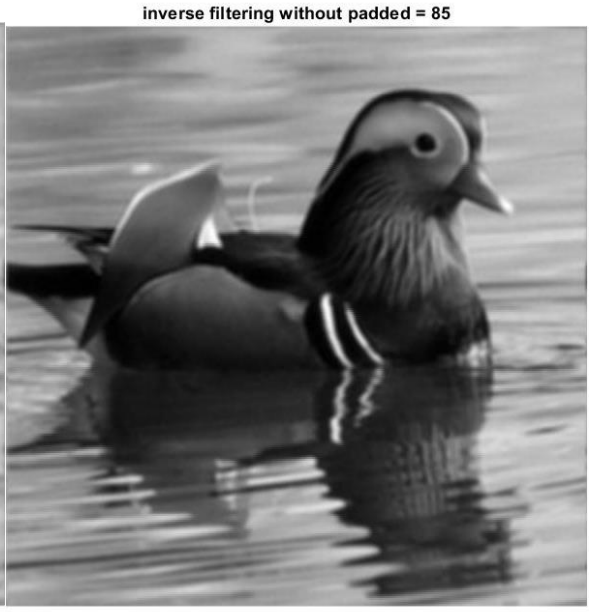


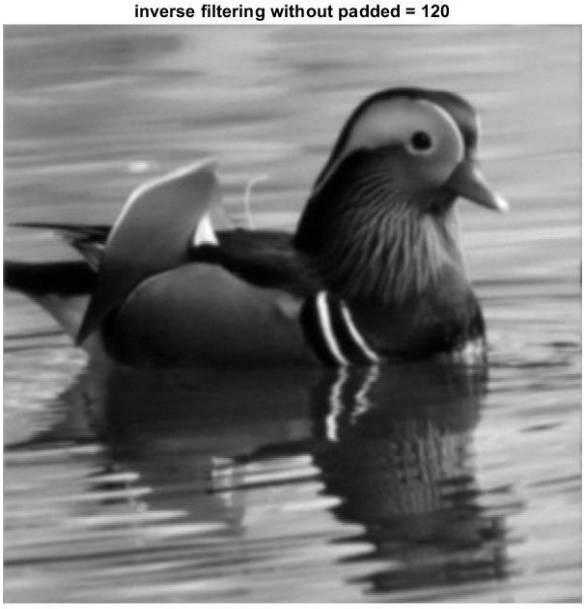
- Figures of the output images using different radii (50, 85, 120) of inverse filtering
  - padded





✧ Without padded





➢ Model parameter k

k = 4.7371e-06

- Source codes

```
clc
clear
%%  Figure of the Fourier magnitude spectrum of the degraded image
degraded_image = im2double(imread('Bird 2 degraded.tif'));
degraded_image_fft = fft2(degraded_image);
degraded_image_shift = fftshift(degraded_image_fft);
degraded_image_fft_magLOG = log(1+abs(degraded_image_shift));
degraded_image_fft_mag = abs(degraded_image_shift);

figure(1);
subplot(2,1,1);
imagesc(degraded_image_fft_mag);
colormap('gray') ;colorbar ; title('Fourier magnitude of the image');
subplot(2,1,2);
imagesc(degraded_image_fft_magLOG);
colormap('gray') ; colorbar ; title('Fourier magnitude of the image in
log');
[m,n] = size(degraded_image);
degraded_image_padded = zeros(2*m,2*n);
degraded_image_padded(1:m,1:n) = degraded_image;
degraded_image_fft = fft2(degraded_image_padded);
degraded_image_shift = fftshift(degraded_image_fft);

ori_image = im2double(imread('Bird 2.tif'));
[mm,nn] = size(ori_image);
ori_image_padded =  zeros(2*m,2*n);
ori_image_padded(1:mm,1:nn) = ori_image;
ori_image_fft = fft2(ori_image_padded);
ori_image_shift = fftshift(ori_image_fft);

degradation_model = zeros(2*m,2*n);
degradation_model(1:2*m,1:2*n) =
degraded_image_fft(1:2*m,1:2*n)./ori_image_fft(1:2*m,1:2*n);
degradation_model_mag = abs(degradation_model);
figure(3);
subplot(2,1,1);
imagesc(degradation_model_mag); colormap('gray') ;
```

```
colorbar ;title('degradation model mag');
degradation_model_mag_log = log(abs(degradation_model));
subplot(2,1,2);
imagesc(degradation_model_mag_log); colormap('gray') ;
colorbar;title('degradation model mag in log');

%% Model parameter k
k=0;
for i = 1:2*m
    for j = 1:2*n
        k = k - ((log(degradation_model(i,j)))^1.2)/(i*i+j*j);
    end
end
k = real(k)/(4*m*n);

%% figures of output images using different radii (50,85,120) of inverse
filtering(padded)
degradation_model_ideal = zeros(2*m,2*n);
Butterworth_LPF_50 = zeros(2*m,2*n);
Butterworth_LPF_85 = zeros(2*m,2*n);
Butterworth_LPF_120 = zeros(2*m,2*n);
distance = zeros(2*m,2*n);

for i = 1:2*m
    for j = 1:2*n
        distance(i,j) = sqrt((i-m).^2 + (j-n).^2);
        aaa = -k*(((i-m).^2+(j-n).^2)^(5/6));
        degradation_model_ideal(i,j) = exp(aaa);
        Butterworth_LPF_50(i,j) = 1/(1+(distance(i,j)/100)^20);
        Butterworth_LPF_85(i,j) = 1/(1+(distance(i,j)/170)^20);
        Butterworth_LPF_120(i,j) = 1/(1+(distance(i,j)/240)^20);
    end
end


fAssume = degraded_image_shift ./ degradation_model_ideal;
```

```
fAssume_50 = fAssume .* Butterworth_LPF_50;
fAssume_85 = fAssume .* Butterworth_LPF_85;
fAssume_120 = fAssume .* Butterworth_LPF_120;

fAssume_50 = real(ifft2(ifftshift(fAssume_50)));
fAssume_85 = real(ifft2(ifftshift(fAssume_85)));
fAssume_120 = real(ifft2(ifftshift(fAssume_120)));

fAssume_50 = fAssume_50-min(fAssume_50(:));
fAssume_85 = fAssume_85-min(fAssume_85(:));
fAssume_120 = fAssume_120-min(fAssume_120(:));

fresult_50 = fAssume_50 ./ max(fAssume_50(:)).*255;
fresult_85 = fAssume_85 ./ max(fAssume_85(:)).*255;
fresult_120 = fAssume_120 ./ max(fAssume_120(:)).*255;

figure(4);
imshow(uint8(fresult_50(1:m,1:n))); title('inverse filtering = 50');
figure(5);
imshow(uint8(fresult_85(1:m,1:n))); title('inverse filtering = 85');
figure(6);
imshow(uint8(fresult_120(1:m,1:n))); title('inverse filtering = 120');

%% figures of output images using diferent radii (50,85,120) of inverse
filtering(without padded)
degraded_image_fft = fft2(degraded_image);
degraded_image_shift = fftshift(degraded_image_fft);
[m,n] = size(degraded_image);
% assume n=1
degradation_model_ideal = zeros(m,n);
Butterworth_LPF_50 = zeros(m,n);
Butterworth_LPF_85 = zeros(m,n);
Butterworth_LPF_120 = zeros(m,n);
distance = zeros(m,n);
```

```
for i = 1:m
    for j = 1:n
        distance(i,j) = sqrt((i-m/2).^2 + (j-n/2).^2);
        aaa = -k*(((i-m/2).^2+(j-n/2).^2)^(5/6));
        degradation_model_ideal(i,j) = exp(aaa);
        Butterworth_LPF_50(i,j) = 1/(1+(distance(i,j)/50)^20);
        Butterworth_LPF_85(i,j) = 1/(1+(distance(i,j)/85)^20);
        Butterworth_LPF_120(i,j) = 1/(1+(distance(i,j)/120)^20);
    end
end

fAssume = degraded_image_shift ./ degradation_model_ideal;

fAssume_50 = fAssume .* Butterworth_LPF_50;
fAssume_85 = fAssume .* Butterworth_LPF_85;
fAssume_120 = fAssume .* Butterworth_LPF_120;

fAssume_50 = real(ifft2(ifftshift(fAssume_50)));
fAssume_85 = real(ifft2(ifftshift(fAssume_85)));
fAssume_120 = real(ifft2(ifftshift(fAssume_120)));

fAssume_50 = fAssume_50-min(fAssume_50(:));
fAssume_85 = fAssume_85-min(fAssume_85(:));
fAssume_120 = fAssume_120-min(fAssume_120(:));

fresult_50 = fAssume_50 ./ max(fAssume_50(:)).*255;
fresult_85 = fAssume_85 ./ max(fAssume_85(:)).*255;
fresult_120 = fAssume_120 ./ max(fAssume_120(:)).*255;

figure(7);
imshow(uint8(fresult_50(1:m,1:n))); title('inverse filtering without
padded = 50');
figure(8);
imshow(uint8(fresult_85(1:m,1:n))); title('inverse filtering without
padded = 85');
figure(9);
imshow(uint8(fresult_120(1:m,1:n))); title('inverse filtering without
padded = 120');
```