

Contents

- [Part\(c\)](#)
- [Part\(d\)](#)

Part(c)

```
close all; clear; clc;
barbara = read(Tiff('barbara.tiff', 'r'));
baby = read(Tiff('baby.tiff', 'r'));

barbBig = barbara(1:512,37:548);
babyBig = baby(201:712,201:712);

baby64_1 = imresize(babyBig,0.125,'nearest');
figure,
subplot(1,2,1), imshow(baby64_1), title("baby64_1");

babylow = filter2(ones(8)/64,babyBig);
baby64_2 = imresize(babylow,0.125,'nearest');
subplot(1,2,2),
% imshow(baby64_2, [0 255]),
imshow(uint8(baby64_2)), title("baby64_2");
```



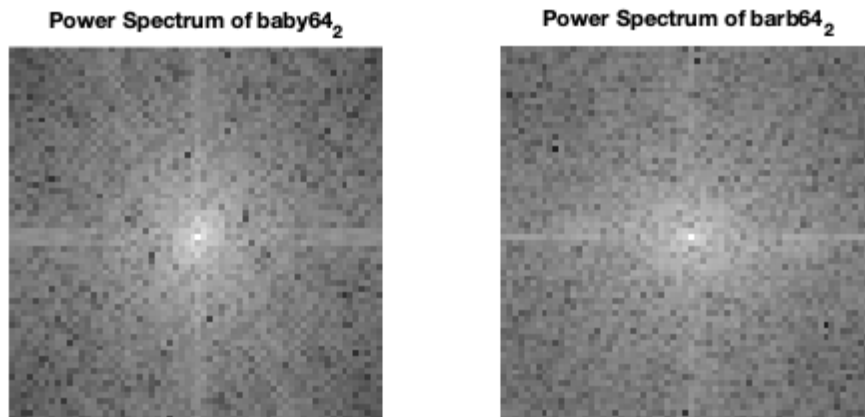
```
barb64_1 = imresize(barbBig,0.125,'nearest');
figure, subplot(1,2,1),
imshow(barb64_1), title("barb64_1")
```

```
barblow = filter2(ones(8)/64,barbBig);  
barb64_2 = imresize(barblow,0.125,'nearest');  
subplot(1,2,2),  
imshow(barb64_2, [0 255]), title("barb64_2");
```



```
baby64fft_1 = fftshift(abs(fft2(baby64_2)).^2);  
baby64spectrum = log(baby64fft_1 + 1);  
figure, subplot(1,2,1),  
imshow(baby64spectrum / max(max(baby64spectrum))),  
title('Power Spectrum of baby64_2');
```

```
barb64fft_1 = fftshift(abs(fft2(barb64_2)).^2);  
barb64spectrum = log(barb64fft_1 + 1);  
subplot(1,2,2),  
imshow(barb64spectrum / max(max(barb64spectrum))),  
title('Power Spectrum of barb64_2');
```



By visually comparing the two 64x64 versions of baby and for barbara, looks like the pre-filtering helped barbara more because of the higher frequency components and aliasing as observed before. barb64_1 looks bad, like around the scarf area where we have vertical lines in the original image with a lot of aliasing because it has high freq which we suppress using low freq filter and thus get a better image barb64_2 baby64_1 and baby64_2 looks identical, however, 1 is still little blurry compared to 2. similarity between them is likely due to the low freq nature of the image. It didn't have aliasing which required the pre-filtering, and thus it didn't make any difference.

Part(d)

Cutoff frequency $\leq 1/2 * \text{sampling frequency}$ sampling frequency = $1/8$ therefore, the cutoff freq is $1/16$

```
barbBig = barbara(1:512,37:548);
babyBig = baby(201:712,201:712);

barbfft_1 = fftshift(abs(fft2(barbBig)).^2);
barbspectrum = log(barbfft_1 + 1);

barbfft_1 = fftshift(abs(fft2(barbBig)).^2);
barbspectrum = log(barbfft_1 + 1);
figure,
I = barbspectrum / max(max(barbspectrum));
rec = insertShape(I,"filled-rectangle",[225 225 64 64], 'Color' , 'red');
imshow(rec)
title('Power Spectrum of barbBig');

% not Aliased Energy
ERM = sum(barbfft_1(225:288,225:288),'all');

% Aliasing Energy
```

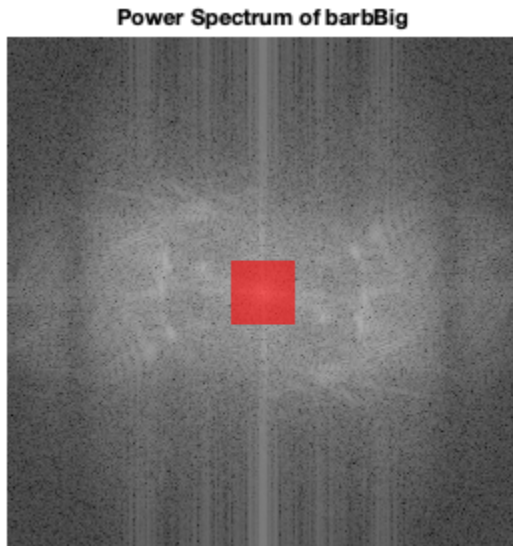
```

EAM = sum(barbfft_1,'all') - ERM;

fprintf('The not Aliased Energy is %i.\n', ERM);
fprintf('The Aliasing Energy %i.\n', EAM);

```

The not Aliased Energy is 1.022746e+15.
The Aliasing Energy 2.233135e+13.



```

% filtered!
barbfft_2 = fftshift(abs(fft2(barblow)).^2);
barbspectrum = log(barbfft_2 + 1);
figure,
I_64_2 = barbspectrum / max(max(barbspectrum));
rec = insertShape(I_64_2,"filled-rectangle",[225 225 64 64], 'Color' , 'red');
imshow(rec)
title('Power Spectrum of barb64 after filtering');

% not Aliased Energy
ER = sum(barbfft_2(225:288,225:288),'all');

% Aliasing Energy
EA = sum(barbfft_2,'all') - ER;

% Aliasing Reduction
aliasing_reduction = (EAM-EA)/EAM;

% Resolution Error
resolution_error = (ERM-ER)/ERM;

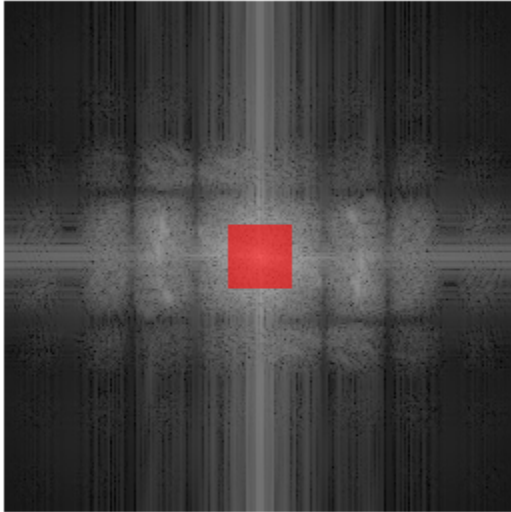
fprintf('The not Aliased Energy after filter is %i.\n', ER);
fprintf('The Aliasing Energy after filter %i.\n', EA);

```

```
fprintf('Aliasing Reduction %i.\n', aliasing_reduction);  
fprintf('Resolution Error %i.\n', resolution_error);
```

The not Aliased Energy after filter is 1.000850e+15.
The Aliasing Energy after filter 2.317592e+12.
Aliasing Reduction 8.962180e-01.
Resolution Error 2.140916e-02.

Power Spectrum of barb64 after filtering



% In the above image of the power spectra, the image where it is aliased and not-aliased is shown. The region inside the inner box with a red boarder is where the spectrum shows part which is not aliased. The region outside is aliased.
