ECE 637 Lab 2 Report

2 – D Random Processes

Xihui Wang

Section 1 – Power Spectral Density of an Image

In this section, we are going to calculate the power spectral density by computing the logarithm of the normalized energy spectrum over different window size of the image. After that, write a better estimate of the power spectral density using 25 non-overlapping image windows of size 64 x 64.

1.1 The Gray Scale Image *img04g.tif*

Fig 1-1 The gray scale image *img04g.tif*

1.2 The Power Spectral Density Plots for Block Sizes of 64 x 64, 128 x 128, 256 x 256

The mesh plots of estimated log power spectral density for block sizes of 64 x 64 (Fig 1-2-1), 128 x 128 (Fig 1-2-2), and 256 x 256 (Fig 1-2-3).

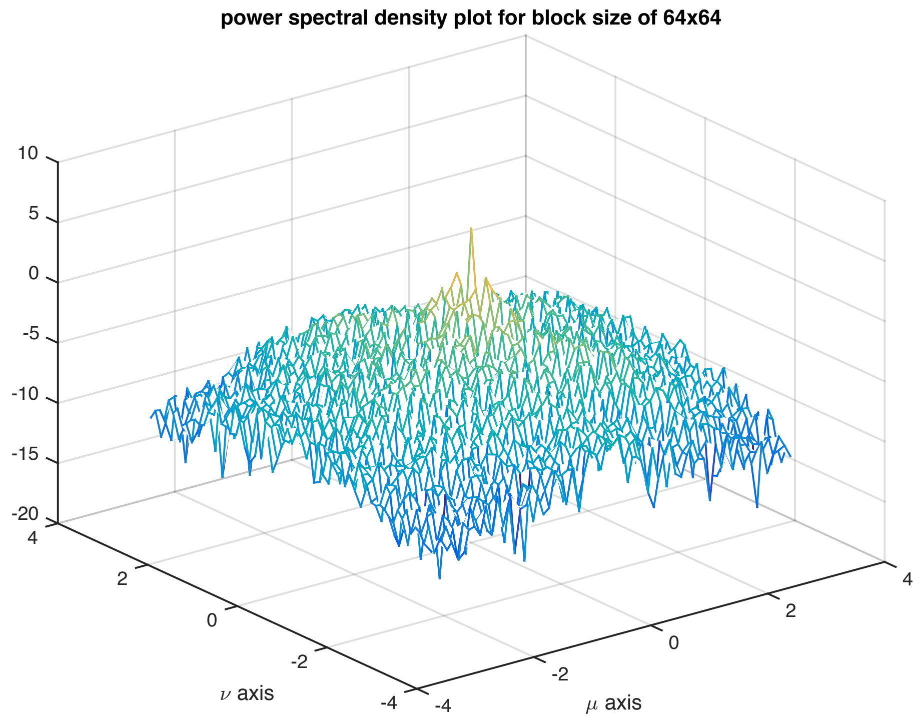

Fig 1-2-1 PSD plot for block size of 64 x 64

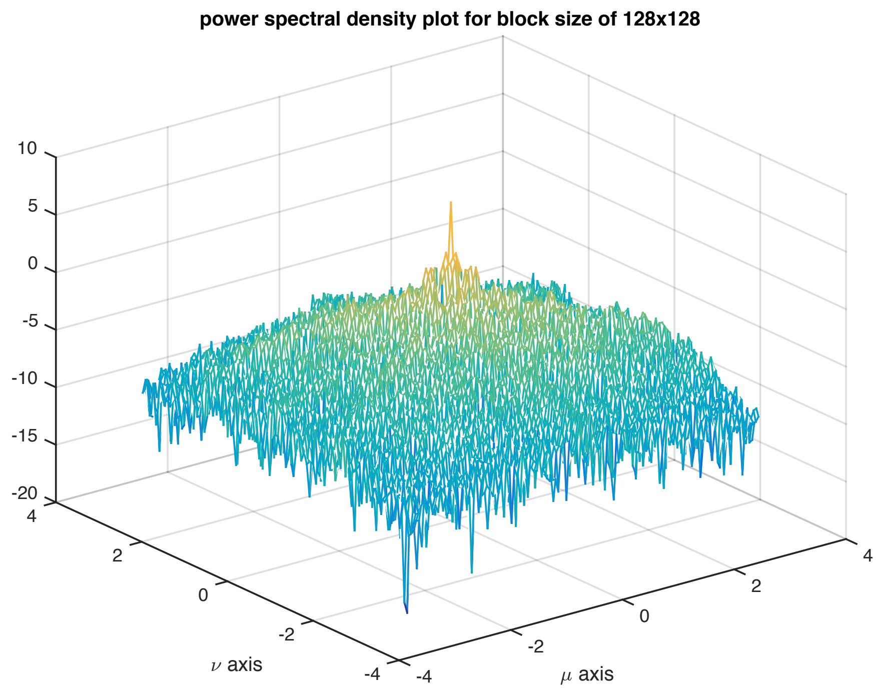

Fig 1-2-2 PSD plot for block size of 128 x 128

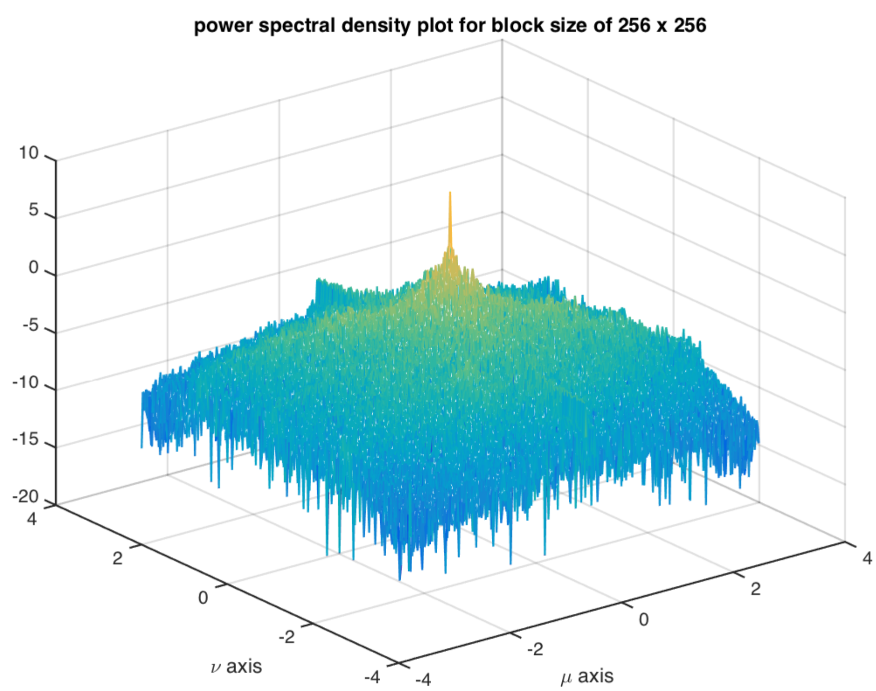

Fig 1-2-3 PSD plot for block size of 256 x 256

1.3 The Improved Power Spectral Density Estimate

A better way to calculate the power spectral density is breaking image into 25 regions, computing the squared DFT magnitude for each block, and averaging blocks to form power spectrum estimate. A mesh plot of the log of the estimated PSD using above algorithm is shown in Fig 1-3.

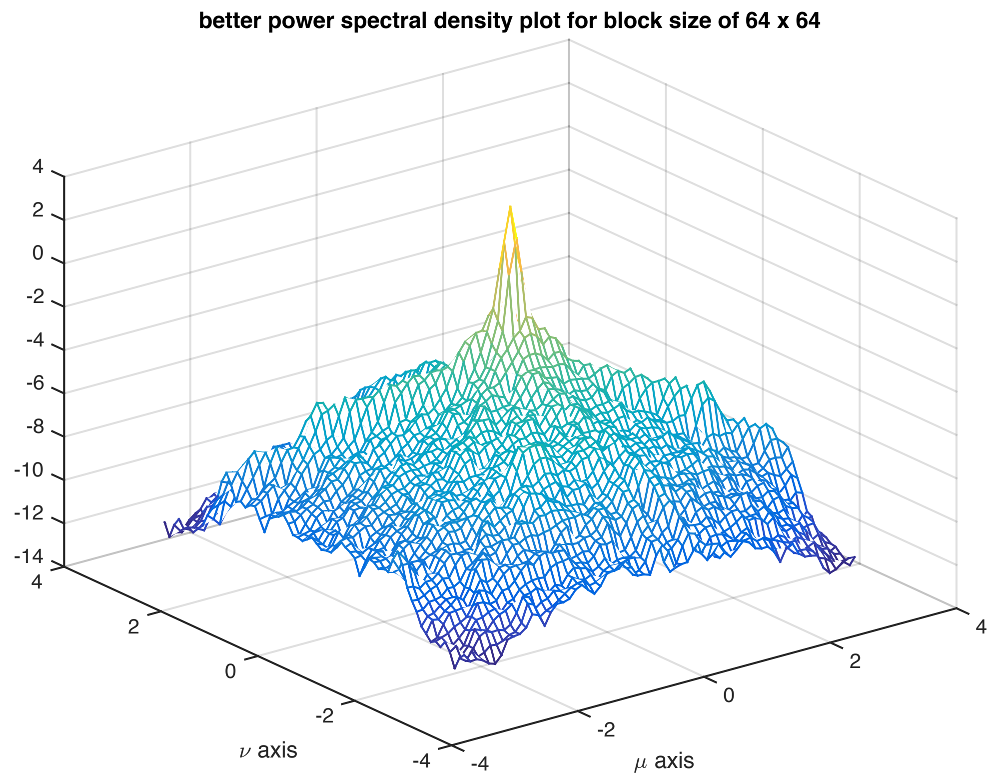

Fig 1-3 The improved PSD estimate for block size of 64 x 64

1.4 The Matlab Code for *BetterSpecAnal.m*

```
function [] = BetterSpecAnal(img)

X = double(img);

% Select an NxN region of the image and store it in the variable "z"
N = 64;

% Multiply each 64 x 64 window by a 2-D separable Hamming window
W = hamming(N)*hamming(N)';

% Break image into K^2 regions each with N pixels
K = 5;
[Xlen, Xwid] = size(X); %512; 768

% center point of X is
len = Xlen/2;
wid = Xwid/2;

% start point (top left corner)
slen = len - K/2*N;
swid = wid - K/2*N;

Z = zeros(N, N);

for k1 = 1:1:K
    for k2 = 1:1:K
        Xtemp = X(((slen+N*(k1-1)):1:(slen+N*k1-1)), ((swid+N*(k2-
1)):1:(swid+N*k2-1)));
        Z = Z + (abs(fft2(Xtemp.*W)).^2) / ((K*N)^2);
```

```
    end
end

% Use fftshift to move the zero frequencies to the center of the plot
Z = fftshift(Z);

% Compute the logarithm of the Power Spectrum.
Zabs = log( Z );



% Plot the result using a 3-D mesh plot and label the x and y axises
properly.

x = 2*pi*((0:(N-1)) - N/2)/N;
y = 2*pi*((0:(N-1)) - N/2)/N;
figure
mesh(x,y,Zabs)
xlabel('\mu axis')
ylabel('\nu axis')
title('better power spectral density plot for block size of 64 x 64')
```

Section 2 – Power Spectral Density of a 2-D AR Process

In this section, we are going to generate a synthetic 2-D autoregressive (AR) process using Matlab, and analyze its power spectral density.

2.1 The Image 255 * (x + 0.5)

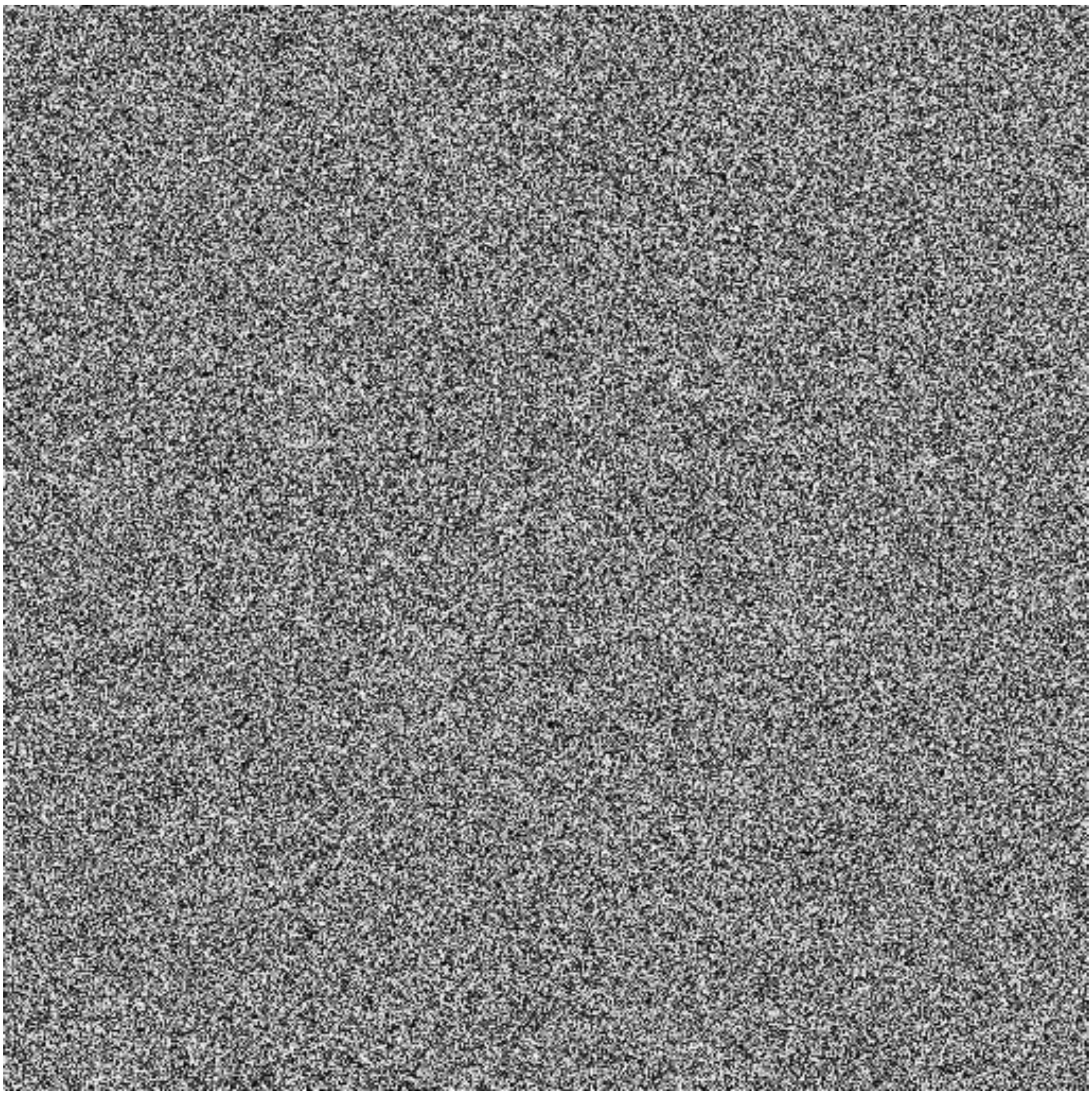

Fig 2-1 Random image generated by 512 x 512 random number each uniformly distributed on the interval [-0.5, 0.5]

2.2 The Image y + 127

The IIR filter transfer function is

$$H(z_1, z_2) = \frac{3}{1 - 0.99z_1^{-1} - 0.99z_2^{-1} + 0.9801z_1^{-1}z_2^{-1}}$$

Since $H(z_1, z_2) = \frac{Y(z_1,z_2)}{X(z_1,z_2)}$, we could get

$$3X(z_1, z_2) = Y(z_1, z_2) - 0.99z_1^{-1}Y(z_1, z_2) - 0.99z_2^{-1}Y(z_1, z_2) + 0.9801z_1^{-1}z_2^{-1}Y(z_1, z_2)$$

It is the same as

$$Y(z_1, z_2) = 3X(z_1, z_2) + 0.99z_1^{-1}Y(z_1, z_2) + 0.99z_2^{-1}Y(z_1, z_2) - 0.9801z_1^{-1}z_2^{-1}Y(z_1, z_2)$$

Inverse Z-transform, we could get

$$y(m, n) = 3x(m, n) + 0.99y(m-1, n) + 0.99y(m, n-1) - 0.9801 * y(m-1, n-1)$$

Then based on above information, display the image of y + 127 in Fig 2-2.

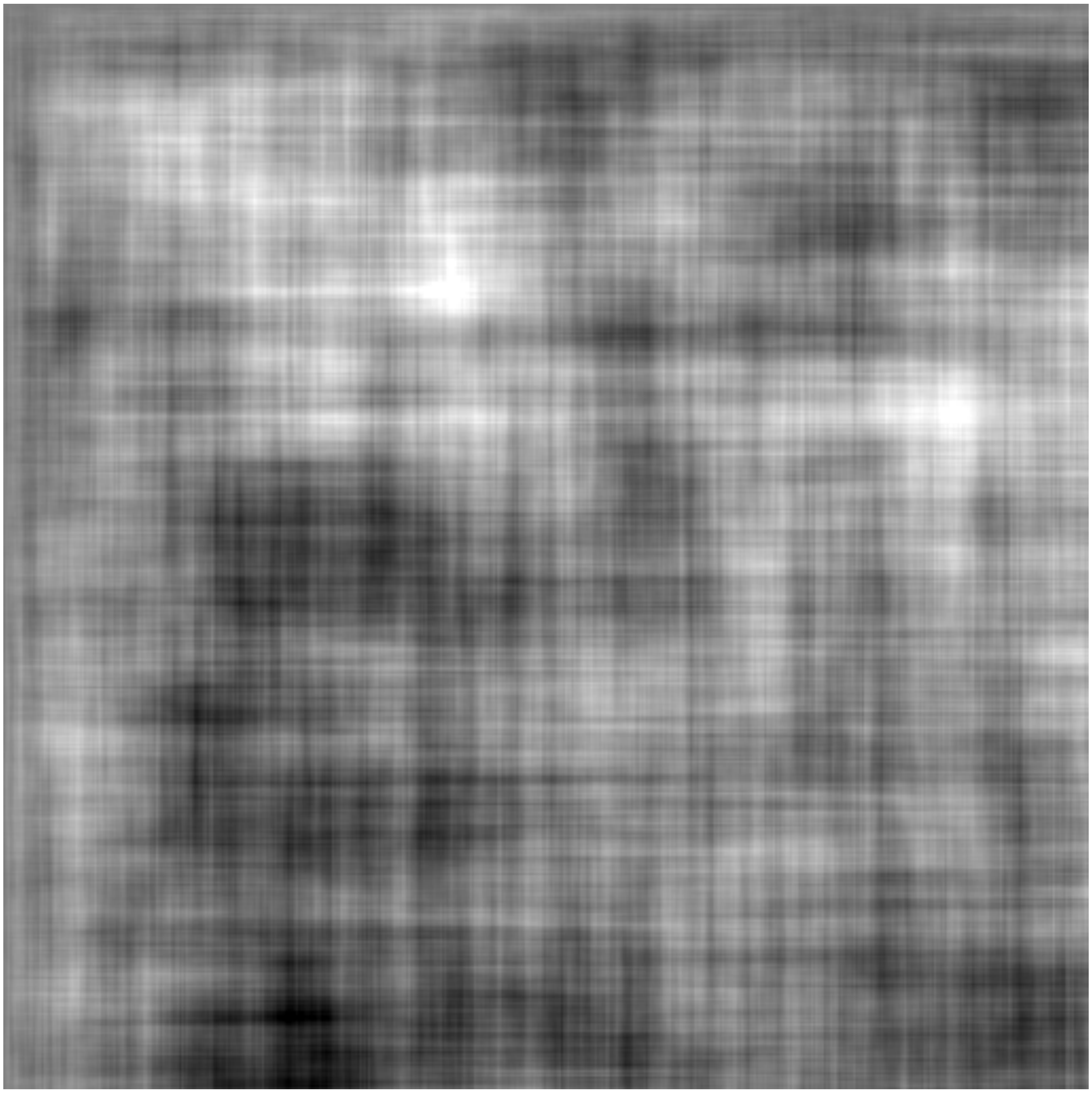

Fig 2-2 The image y+127

2.3 A Mesh Plot of The Function $\log S_y(e^{j\mu}, e^{j\nu})$

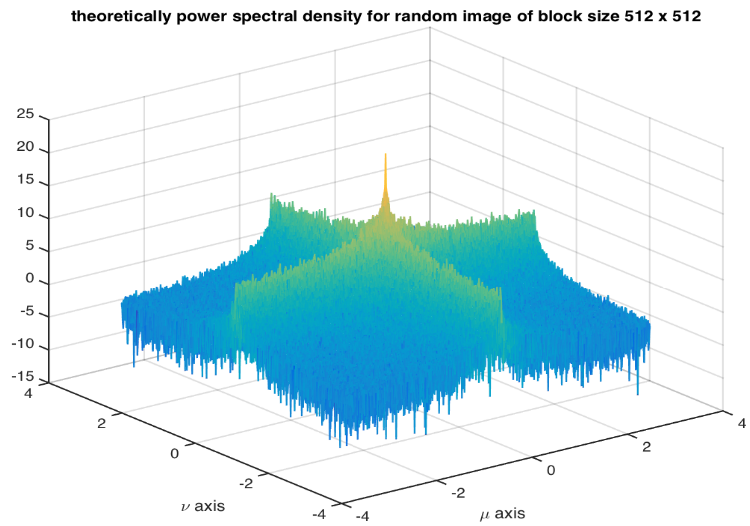

Fig 2-3 Mesh Plot of Theoretical PSD

2.4 A mesh plot of the log of the estimated power spectral density of y using *BetterSpecAnal(y)*

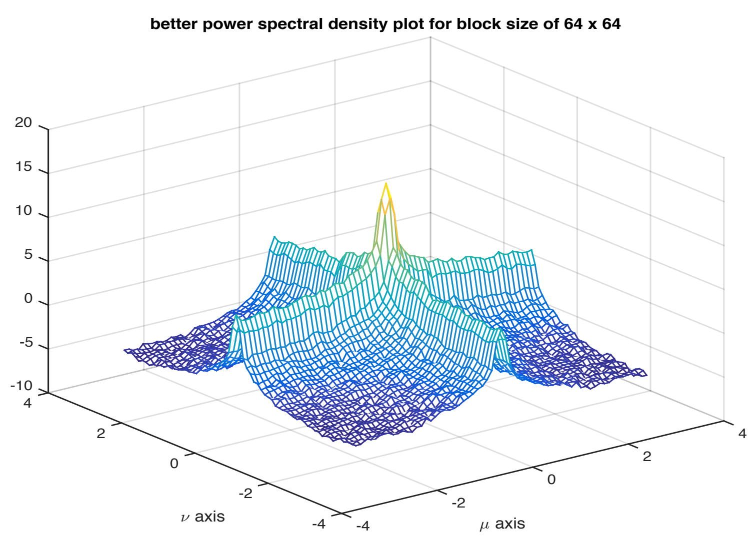

Fig 2-4 Mesh Plot of Better Estimated PSD

2.5 Matlab Code for 2-D AR Process

```
% Generate a 512 x 512 image, x, with independent random numbers
% each uniformly distributed on the interval [-0.5, 0.5]
x = rand(512) - 0.5;
x_scaled = 255*(x+0.5);

imwrite(uint8(x_scaled),'random.tif');
y = zeros(513);
for m=1:1:512
    for n=1:1:512
        y(m+1,n+1) = 3*x(m,n)+0.99*y(m,n+1)+0.99*y(m+1,n)-0.9801*y(m,n);
    end
end
y = y(2:1:513,2:1:513) + 127;
imwrite(uint8(y),'randomy127.tif');

BetterSpecAnal(y);

N = 512;

% Compute the power spectrum for the NxN region
Z = (1/N^2)*abs(fft2(y)).^2;

% Use fftshift to move the zero frequencies to the center of the plot
Z = fftshift(Z);

% Compute the logarithm of the Power Spectrum.
Zabs = log( Z );
```

```
% Plot the result using a 3-D mesh plot and label the x and y axises
properly.
x = 2*pi*((0:(N-1)) - N/2)/N;
y = 2*pi*((0:(N-1)) - N/2)/N;
figure
mesh(x,y,Zabs)
xlabel('\mu axis')
ylabel('\nu axis')
title('theoretically power spectral density for random image of block
size 512 x 512')
```