ASSIGNMENT 2

Image Processing and Application - 9804

JEWEL MAHMUD NIMUL SHAMIM
201890846

Answer to the question no: 1

MATLAB Code

```
clc;
clear all;
close all;
% Read image
image = imread('moonlanding.png');
imshow(image)
%Convert image from spatial domain to Frequency Domain
FD = fft2(image);
FS = fftshift(log(1+abs(FD)));
%Find the maximum frequency in S
FSmax = max(FS(:));
imshow(FS,[]) % Display The figure in frequency domain
disp(FSmax)
findex = 1;
F = getframe;
imwrite(F.cdata, 'spectrum.png')
```

OUTPUT

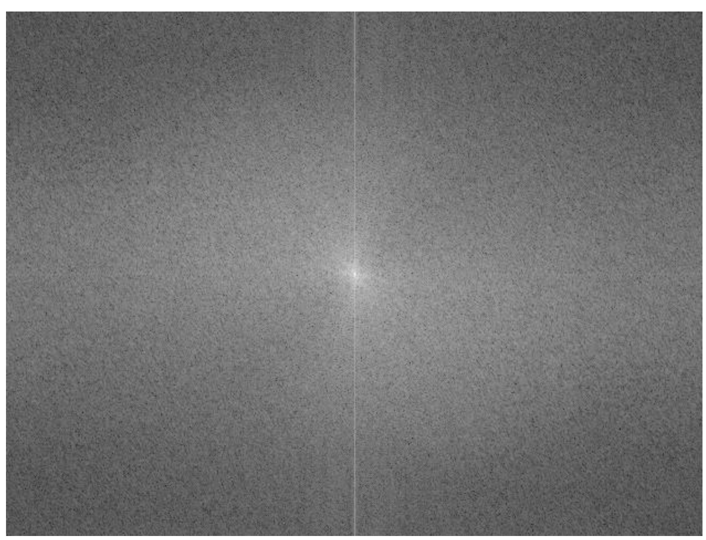
```
Command Window
    16.9610

fx >>
```

The maximum value of the frequency spectrum: **16.9610**

Answer to the question no: 2

The frequency spectrum obtained from the Question: 1

OUTPUT

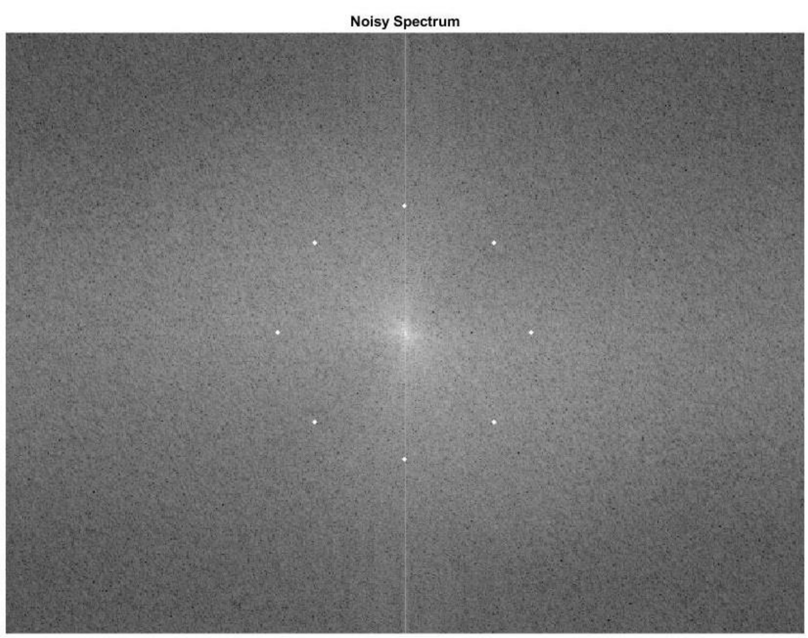


Answer to the question no: 3

MATLAB Code

```
clc;
clear all;
close all;
image = imread('moonlanding.png');
F = fft2(image); %Convert spatial domain to Frequency Domain
FS = fftshift(log(1+abs(F))); %Calculate the DFT.
FSmax = max(FS(:)); %Find the maximum frequency
[A B] = size(FS);
CRDr = 100;
CRx = B/2;
CRy = A/2;
th = 0:pi/5:2*pi;
% Display The figure in frequency domain
imshow(FS,[])
hold on
th = 0:pi/4:2*pi;
xunit = CRDr * cos(th) + CRx;
yunit = CRDr * sin(th) + CRy;
%scatter(xunit, yunit, 20,'filled');
scatter(xunit,yunit,1,'MarkerEdgeColor',[1 1 1], 'LineWidth', 2.0)
hold off
frame_index = 1;
F = getframe ;
imwrite(F.cdata, 'data.png')
```

OUTPUT

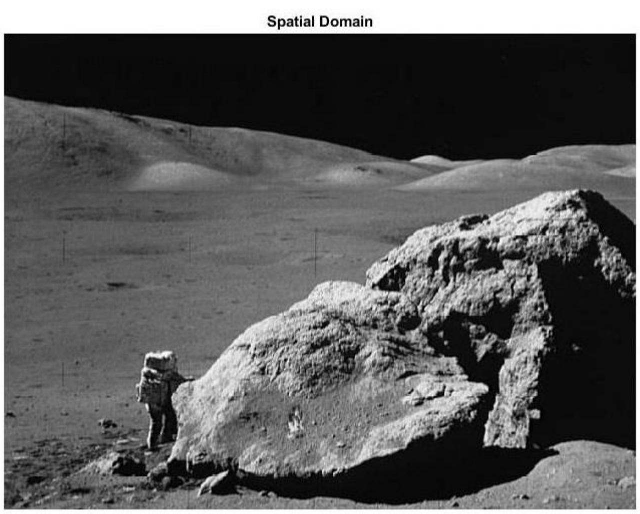


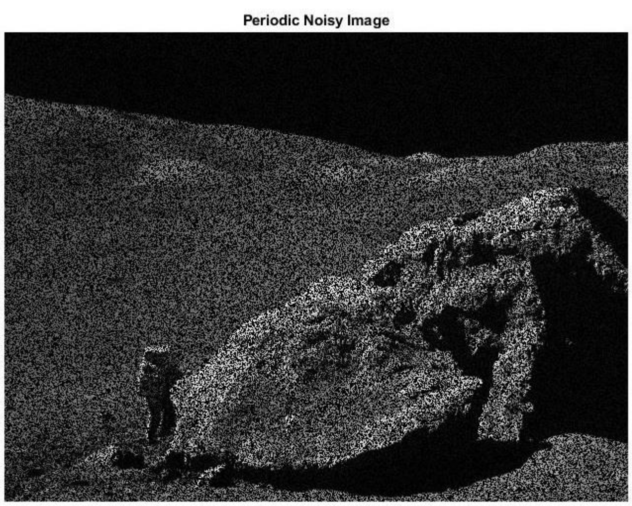
Answer to the question no: 4

MATLAB Code

```
clc;
clear all;
close all;
image = imread("moonlanding.png");
FFT = fft2(image); % 2D DFT using FFT

spatial = imresize(ifft2(FFT),size(image));
figure('Name',"Spatial Domain")
imshow(spatial,[]);
title("Spatial Domain")

noise = 0.5;
size = rand(size(image));
noisy = find(size < noise/2);
image(noisy) = 0;
noisy = find(size >= noise/2 & size < noise);
image(noisy) = 1;

figure('Name',"Noisy")
imshow(image)
title("Periodic Noisy Image")
```

OUTPUT





Comments:
After adding the periodic noise, the image is unclear and loses a lot of information from the image. Also, the brighter area has more noise.

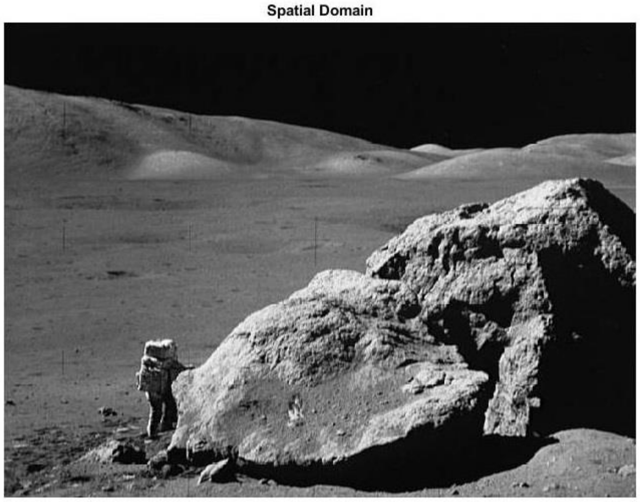
Answer to the question no: 5

MATLAB Code

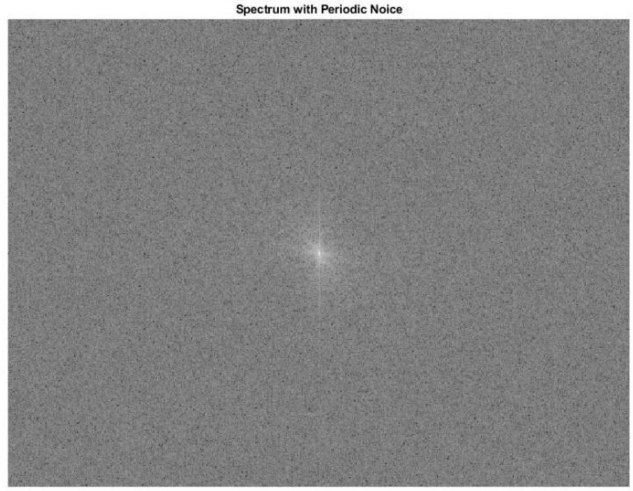
```
clc;
clear all;
close all;
image = (imread("moonlanding.png"));
FFT = fft2(image); %perform 2D DFT using FFT algorithmfigure,
%FS = fftshift(log(1+abs(F)));
spatial = imresize(ifft2(FFT), size(image))
points = 0.5;
x = rand(size(image));
noisy = find(x < points/2);
image(noisy) = 0;
noisy = find(x >=points/2 & x < points);
image(noisy) = 1;

FFT = fft2(image);
ShiftedImage = fftshift(log(1+abs(FFT)));
imshow(ShiftedImage,[]);
title('Spectrum with Periodic Noice');
```

OUTPUT

Spatial Domain



Spectrum with Periodic Noice



Comments:

Manually created spectrum and from the periodic noisy image spectrum are not the same. Also The frequency domain without noise compared to brighter on this image.

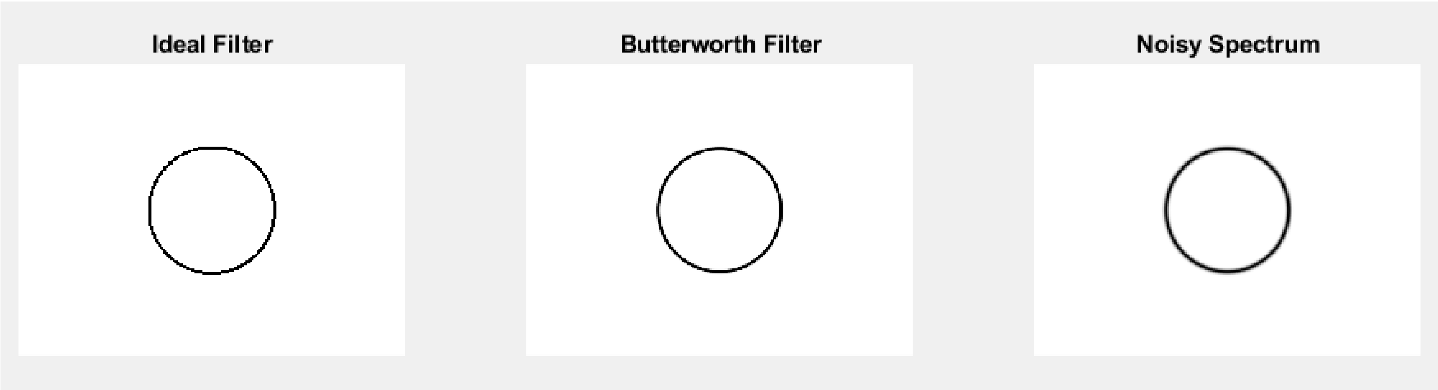
Answer to the question no: 6

MATLAB Code

```
function y = distmatrix(M,N)
    u = 0:(M - 1);
    v = 0:(N - 1);

    ind_u = find(u > M/2);
    u(ind_u) = u(ind_u) - M;
    ind_v = find(v > N/2);
    v(ind_v) = v(ind_v) - N;

    [V, U] = meshgrid(v, u);

    %calculate distance matrix
    y = sqrt((U .^ 2) + (V .^ 2));
end



function b = ideal_lp(M,N,cut_off)
    dist = distmatrix(M,N);
    H = zeros(M,N);
    ind = dist <= cut_off;
    H(ind) = 1;
    b = double(H);
end


function b = ideal_hp(M,N,cut_off)
    dist = distmatrix(M,N);
    H = ones(M,N);
    ind = dist<= cut_off;
    H(ind) = 0;
    b = double(H);
end


function b = ideal_br(M,N,cut_off1,cut_off2)
    if cut_off1 > cut_off2
        h_cut_off = cut_off1;
        l_cut_off = cut_off2;
    else
        h_cut_off = cut_off2;
        l_cut_off = cut_off1;
    end
    H1 = ideal_hp(M,N,l_cut_off);
    H2 = ideal_lp(M,N,h_cut_off);
    Hbr = H2 - H1;
    ind = Hbr == -1;
    Hbr(ind) = 1;
    b = Hbr;
end
```

```
function b = gaussian_br(M,N,cut_off1,cut_off2)
if cut_off1 > cut_off2
    h_cut_off = cut_off1;
    l_cut_off = cut_off2;
else
    h_cut_off = cut_off2;
    l_cut_off = cut_off1;
end
dist = distmatrix(M,N);
W = h_cut_off - l_cut_off;
D0 = l_cut_off;
Hbr = 1 - exp((-1/2)*(((dist.^2)-(D0^2))./ (dist*W)).^2);
b = Hbr;
end


function b = butter_br(M,N,cut_off1,cut_off2, ord)
    if cut_off1 > cut_off2
        h_cut_off = cut_off1;
        l_cut_off = cut_off2;
    else
        h_cut_off = cut_off2;
        l_cut_off = cut_off1;
    end
    W = h_cut_off - l_cut_off;
    D0 = l_cut_off;
    dist = distmatrix(M,N);
    Hbr = 1 ./ (1+((dist*W)./((dist.^2)-(D0^2))).^(2*ord));
    b = Hbr;
end
```

OUTPUT

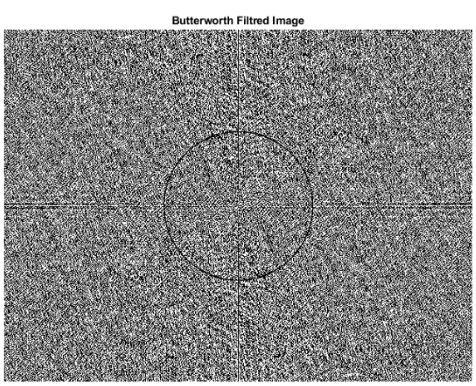


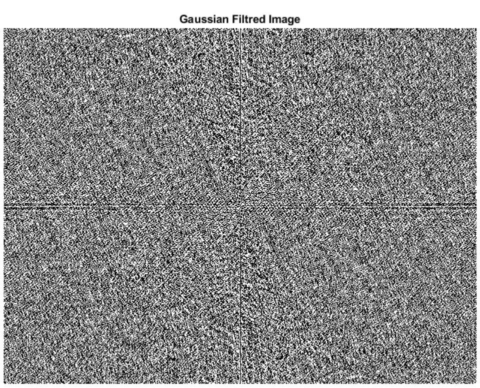
Answer to the question no: 7

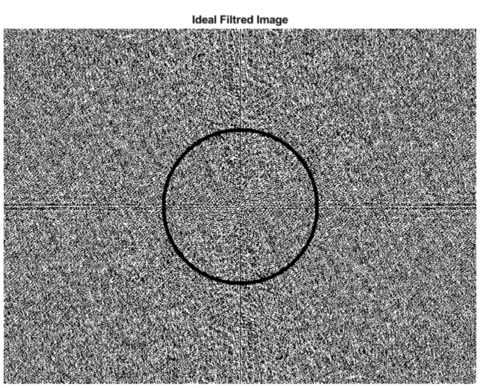
MATLAB Code

```
clc
clear all
close all
image=imread('moonlanding.png');
image = double(image);
imafft = fft2(image);
FS = fftshift(imafft);
% % Fourier Spectrum of Image
s = size(image); ma=max(max((FS)));
maxium = 0.5*sqrt(s(1)^2+s(2)^2);
cutoff1 = maxium*30;
cutoff2 = maxium*120;

butterworth_filter = fftshift(butter_br(474,630,100,105,4));
gussian_filter = fftshift(gaussian_br(s(1),s(2),cutoff1,cutoff2));
Ideal_filter = fftshift(ideal_br(s(1),s(2),100,105));

butter = (immultiply(real(FS),butterworth_filter));
gussian = (immultiply(real(FS),gussian_filter));
ideal = (immultiply(real(FS),Ideal_filter));
figure(1);
imshow(butter)
title("Butter Filtered Image")
figure(2);
imshow(gussian)
title("Gussian Filtered Image")
figure(3);
imshow(ideal)
title("Ideal Filtered Image")
```

OUTPUT





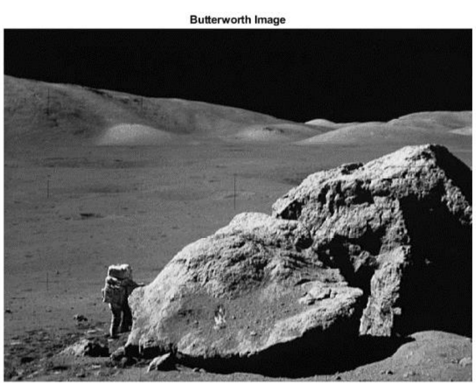


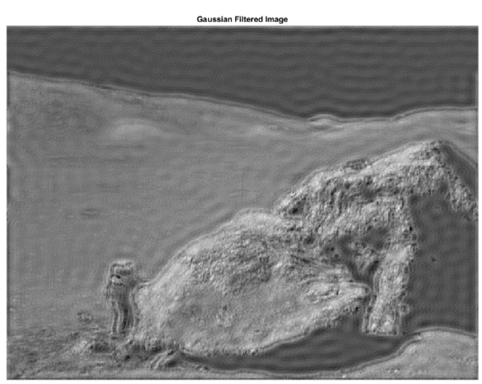
Answer to the question no: 8

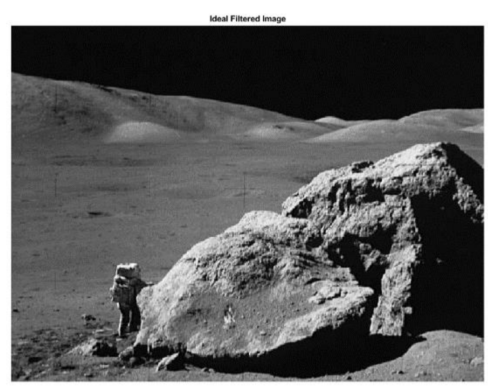
MATLAB Code

```
clc
clear all
close all
image = imread('moonlanding.png');
image = double(image);
figure(1);
imshow(image,[]);
title('Original Image');
imafft = fftshift(fft2(fftshift(image)));
% Fourier Spectrum of Image
imafft2 = fft2(image);
imafft3 = fftshift(imafft2);
size = size(image); ma=max(max((imafft)));
butterworth_filter = fftshift(butter_br(474,630,100,105,4));
gussian_filter = fftshift(gaussian_br(s(1),s(2),cutoff1,cutoff2));
Ideal_filter = fftshift(ideal_br(s(1),s(2),100,105));
butter = (immultiply(real(FS),butterworth_filter));
gussian = (immultiply(real(FS),gussian_filter));
ideal = (immultiply(real(FS),Ideal_filter));
imafft=imafft.*z/255;
image_out = fftshift(ifft2(fftshift(imafft)));
image_out =image_out-image; image_out = 1-image_out;
figure(1); imshow(image_out,[]);
title(Butterworth Filtered Image');
figure(2); imshow(image_out,[]);
title('Gaussian Filtered Image');
figure(3); imshow(image_out,[]);
title(Ideal Filtered Image');
```

OUTPUT







Answer to the question no: 9

Band rejects filters are applied to provide a large band of frequencies while rejecting a small band of frequencies. And also filter out in the selected region image by canceling out the frequencies.

From the output of Butterworth, Gaussian, and ideal filter, we can say that the idea band-reject filter produces a sharper image. The Gaussian band-reject filter provides a heavy, darker image and filtered out periodic noise.