Image Processing and Signal - UsingMatlab

November 29

2024

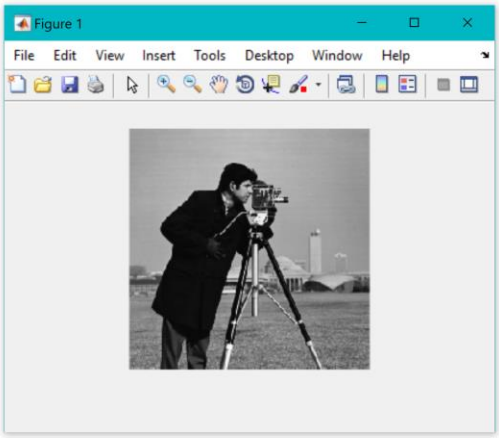
By: Aya Hussain Merai
Class: B

All Programs
Course1

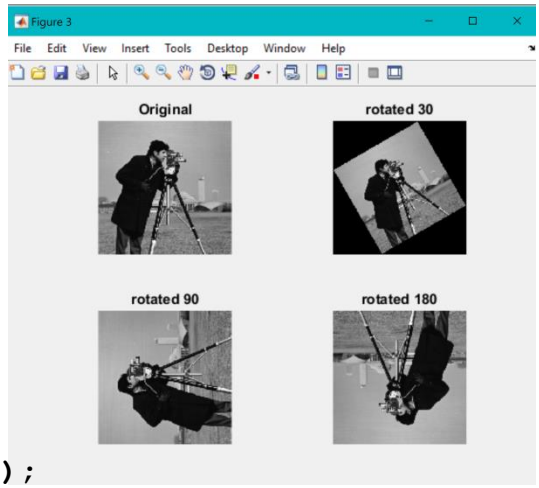
Lab 1

```
clc;
clear;
% Program1
% Read the information and the data of the image,
% and also display the data and show the image.

info = imfinfo('cameraman.tif');
% This ^^^^^^^ reads all the information of the image
disp(info);

f = imread('cameraman.tif');
% Reads all the image's pixels
imshow(f);
% ^^^^ Shows the image
```

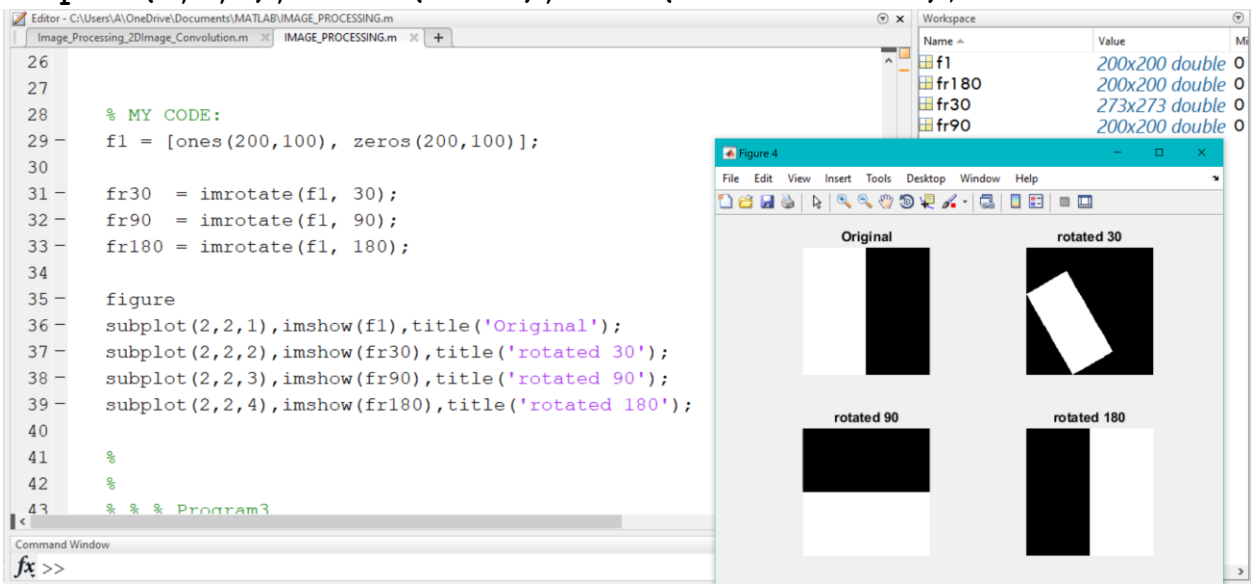
```
Command Window
                    Filename: 'C:\Program Files\MATLAB\MATLAB Production Server\R2015a...'
                FileModDate: '04-Dec-2000 10:57:54'
                    FileSize: 65240
                      Format: 'tif'
                FormatVersion: []
                       Width: 256
                      Height: 256
                    BitDepth: 8
                   ColorType: 'grayscale'
             FormatSignature: [77 77 0 42]
                   ByteOrder: 'big-endian'
              NewSubFileType: 0
               BitsPerSample: 8
                 Compression: 'PackBits'
    PhotometricInterpretation: 'BlackIsZero'
                StripOffsets: [8 8262 16426 24578 32492 40499 48599 56637]
             SamplesPerPixel: 1
                RowsPerStrip: 32
             StripByteCounts: [8254 8164 8152 7914 8007 8100 8038 8235]
                  XResolution: 72
                  YResolution: 72
              ResolutionUnit: 'Inch'
                    Colormap: []
          PlanarConfiguration: 'Chunky'
                  TileWidth: []
                 TileLength: []
                 TileOffsets: []
              TileByteCounts: []
                 Orientation: 1
                   FillOrder: 1
            GrayResponseUnit: 0.0100
              MaxSampleValue: 255
              MinSampleValue: 0
                Thresholding: 1
                      Offset: 64872
          ImageDescription: 'This image is distributed by The MathWorks, Inc. with
  p...'
```



```
% Program2

% Display the image in the figure, and rotate it by angle 30,90,180 Then
Display all images in one figure used subplot

clc;
clear;
f1 = imread('cameraman.tif');
fr30  = imrotate(f1, 30);
fr90  = imrotate(f1, 90);
fr180 = imrotate(f1, 180);

figure
subplot(2,2,1),imshow(f1),title('Original');
subplot(2,2,2),imshow(fr30),title('rotated 30');
subplot(2,2,3),imshow(fr90),title('rotated 90');
subplot(2,2,4),imshow(fr180),title('rotated 180');
```

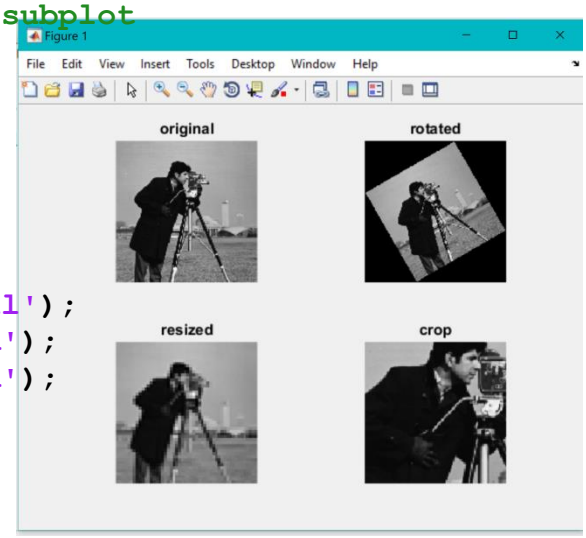



```
% program3

% Create binary image and rotate it by 30, 90, 180
% Then display all images in one figure used subplot

f1 = [ones(200,100), zeros(200,100)];

fr30  = imrotate(f1, 30);
fr90  = imrotate(f1, 90);
fr180 = imrotate(f1, 180);

figure
subplot(2,2,1),imshow(f1),title('Original');
subplot(2,2,2),imshow(fr30),title('rotated 30');
subplot(2,2,3),imshow(fr90),title('rotated 90');
subplot(2,2,4),imshow(fr180),title('rotated 180');
```

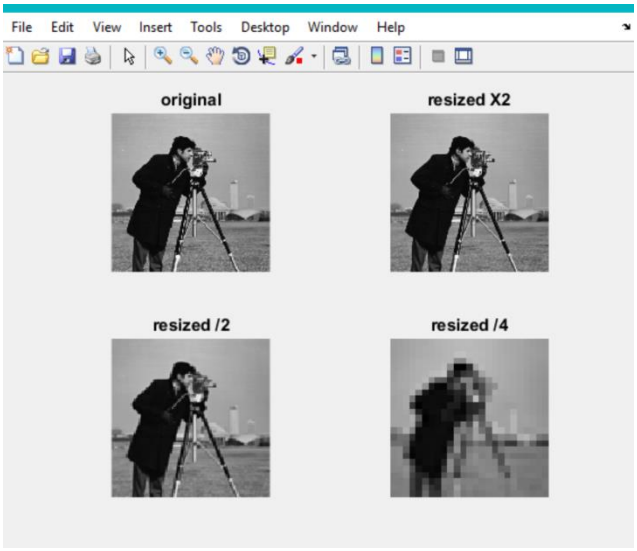


Lab 2

```
clc;
clear;
% program1

% Display the image in figure, and rotate, resize, and crop the image.
% Then display all images in one figure used subplot

f1 = imread('cameraman.tif');
fr = imrotate(f1,30);
fs = imresize(f1,[50 50]);
fc = imcrop(f1,[50 50 100 100]);

figure
     subplot(2,2,1),imshow(f1),title('original');
     subplot(2,2,2),imshow(fr),title('rotated');
     subplot(2,2,3),imshow(fs),title('resized');
     subplot(2,2,4),imshow(fc),title('crop');
```




```
clc;
clear;
% program2

% resize the image by using the following instruction
% Then display all images in one figure used subplot

f1 = imread('cameraman.tif');
fs1 = imresize(f1,[400 400]);
fs2 = imresize(f1,[100 100]);
fs3 = imresize(f1,[25 25]);

figure
subplot(2,2,1),imshow(f1),title('original');
subplot(2,2,2),imshow(fs1),title('resized X2');
subplot(2,2,3),imshow(fs2),title('resized /2');
subplot(2,2,4),imshow(fs3),title('resized /4');
```

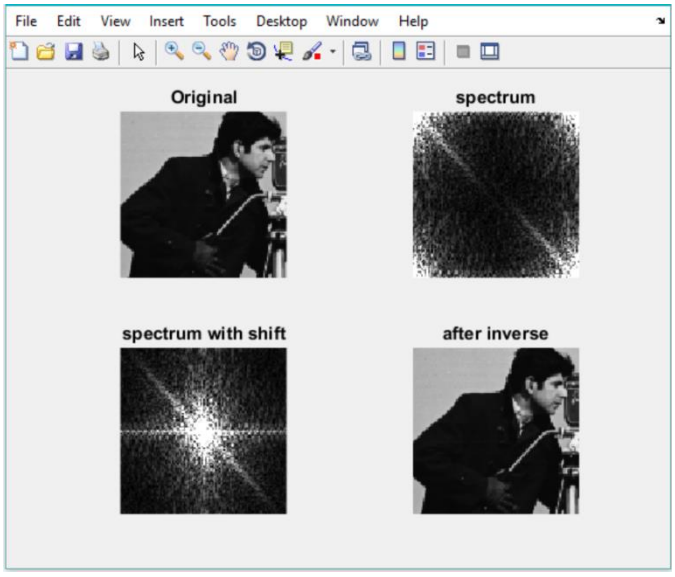


```
% program1
clc;
clear;

% Write Matlab program for Discrete Fourier Transform
% (1 Dimension) of the following values ( samples ):
f = [2 3 4 4];

N = length(f);

F = [];
for u = 1:N
    F(u) = 0;
    for x = 1:N
      F(u) = F(u) + f(x) * exp(-2j * pi * (u-1) * (x-1) / N);
    end
end
F = F / N;

disp('DFT:');
disp(F);

spect = sqrt(real(F).^2 + imag(F).^2);
disp('Spect:');
disp(spect);


% program2

% Write Matlab program for Inverse of Discrete Fourier Transform
% (1Dimension) and find the spectrum of these pixels:

f_inv = zeros(1, N);
for u = 1:N
    for v = 1:N
      f_inv(u) = f_inv(u) + F(v) * exp(2j * pi * (u-1) * (v-1) / N);
    end
end

disp('DFT Inverse:');
disp(round(f_inv));
```

```
Command Window
  DFT:
     3.2500 + 0.0000i  -0.5000 + 0.2500i  -0.2500 - 0.0000i  -0.5000 - 0.2500i

  Spect:
      3.2500    0.5590    0.2500    0.5590

  DFT Inverse:
       2     3     4     4
```

Lab 4

```
clc;
clear;
% Apply Discrete Fourier Transform (1 Dimension) to the image

f = double(imcrop(imread('cameraman.tif'),[40 40 99 99]));

[r, c] = size(f);
f = reshape(f, 1, r*c);
%   ^^^^^^^ convert image dimensions

N = length(f);
F = zeros(1, N);

for u = 1: N
    F(u) = 0;
    for x = 1: N
        F(u) = F(u) + f(x) * exp(-2j * pi * (u-1) * (x-1) / N);
    end
end

F = F / N;
spect = abs(F);

fnew = zeros(1, N);

for x = 1: N
    fnew(x) = 0;
    for u = 1: N
        fnew(x) = fnew(x) + F(u) * exp(2j * pi * (u-1) * (x-1)/ N);
    end
end
fnew = round(fnew);
```




```
f_original = reshape(f, r, c);
f2d_spect = reshape(spect, r, c);
Fshift = fftshift(f2d_spect);
f_inv = reshape(fnew, r, c);

figure
      subplot(2,2,1),imshow(uint8(f_original)),title('Original');
      subplot(2,2,2),imshow(f2d_spect),title('spectrum');
      subplot(2,2,3),imshow(Fshift),title('spectrum with shift');
      subplot(2,2,4),imshow(uint8(f_inv)),title('after inverse');
```

Lab 5

```
% Program1
% Apply DFT (2 Dimension - General) to the following points
f = [0 0 1 0;
     2 0 0 0;
     0 3 0 0;
     0 1 0 2];
[M, N] = size(f);
j = sqrt(-1);
F = zeros(M, N);

for u = 1: M
 for v = 1: N
  F(u,v) = 0;
  for x = 1: M
   for y = 1: N
    F(u,v)=F(u,v)+f(x,y)*exp(-2j*pi*((u-1)*(x-1)/M+(v-1)*(y-1)/N));
   end
  end
 end
end
F = F / (M*N);
disp('DFT:');
disp(F);

spect = abs(F);
disp('Spect:');
disp(spect);
```

Command Window

```
DFT:
   0.5625 + 0.0000i   0.0625 - 0.1250i  -0.1875 - 0.0000i   0.0625 + 0.1250i
  -0.1250 + 0.0625i  -0.1250 + 0.0625i   0.2500 - 0.3125i   0.0000 - 0.3125i
  -0.0625 - 0.0000i  -0.1875 - 0.2500i  -0.0625 + 0.0000i  -0.1875 + 0.2500i
  -0.1250 - 0.0625i  -0.0000 + 0.3125i   0.2500 + 0.3125i  -0.1250 - 0.0625i

Spect:
    0.5625    0.1398    0.1875    0.1398
    0.1398    0.1398    0.4002    0.3125
    0.0625    0.3125    0.0625    0.3125
    0.1398    0.3125    0.4002    0.1398

Inverse:
     0     0     1     0
     2     0     0     0
     0     3     0     0
     0     1     0     2
```

```
% INVERTION
fnew = zeros(1, N);
for x = 1: M
 for y = 1: N
  fnew(x,y) = 0;
  for u = 1: M
   for v = 1: N
    fnew(x,y)=fnew(x,y)+F(u,v)*exp(2j*pi*((u-1)*(x-1)/M+(v-1)*(y-1)/N));
   end
  end
 end
end
fnew = round(fnew);
disp('Inverse: ');
disp(fnew);
```

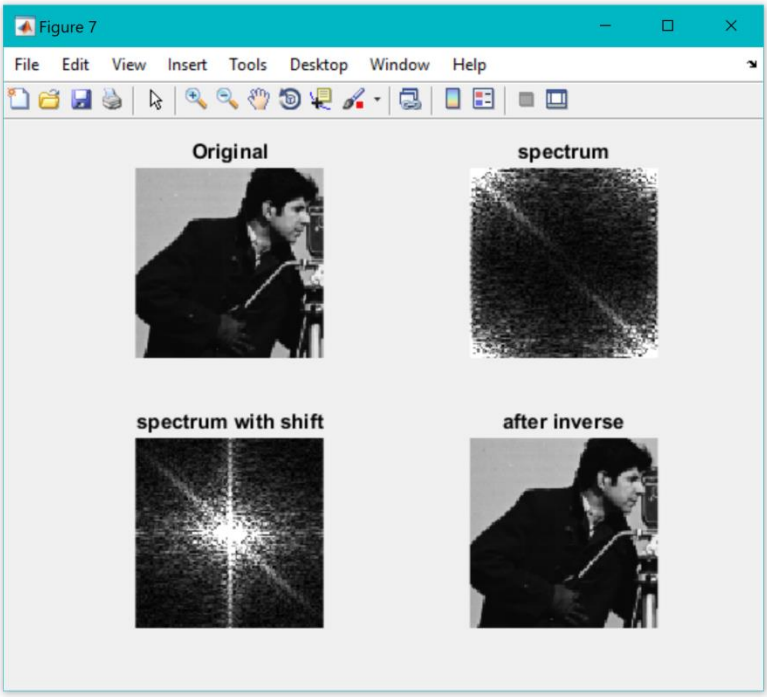
```
clc;
clear;
% Program 2
% Apply DFT (2 Dimension Square) to the following points
f = [0 0 1 0;
     2 0 0 0;
     0 3 0 0;
     0 1 0 2];
[M, N] = size(f);
F = zeros(N);
for u = 1: N
 for v = 1: N
  F(u,v) = 0;
  for x = 1: N
   for y = 1: N
    F(u,v) = F(u,v)+f(x,y)*exp(-2j*pi*(((u-1)*(x-1)+(v-1)*(y-1))/N));
   end
  end
 end
end

F = F / (N);

disp('DFT:');
disp(F);

spect = abs(F);
disp('Spect:');
disp(spect);

% INVERTION
fnew = zeros(N);


for x = 1: N
 for y = 1: N
  fnew(x,y) = 0;
  for u = 1: N
   for v = 1: N
    fnew(x,y) = fnew(x,y)+F(u,v)*exp(2j*pi*((u-1)*(x-1)+(v-1)*(y-1))/N);
   end
  end
 end
end

fnew = round(fnew/N);

disp('Inverse: ');
disp(fnew);
```

Editor - C:\Users\A\OneDrive\Documents\MATLAB\Lab5_ImageProcessing_FourerSQ.m

mmand Window

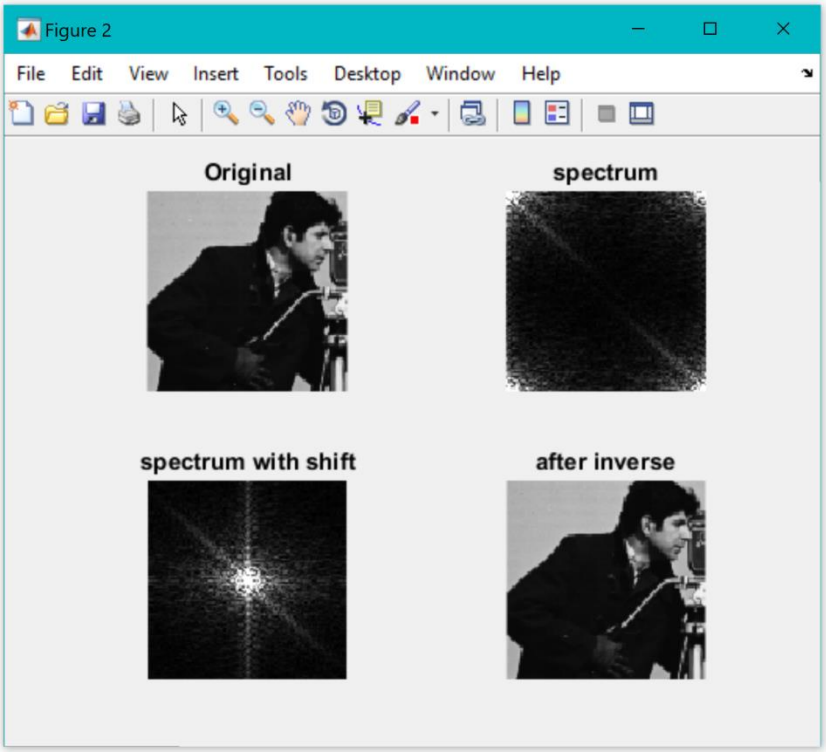
```
DFT:
   2.2500 + 0.0000i   0.2500 - 0.5000i  -0.7500 - 0.0000i   0.2500 + 0.5000i
  -0.5000 + 0.2500i  -0.5000 + 0.2500i   1.0000 - 1.2500i   0.0000 - 1.2500i
  -0.2500 - 0.0000i  -0.7500 - 1.0000i  -0.2500 + 0.0000i  -0.7500 + 1.0000i
  -0.5000 - 0.2500i  -0.0000 + 1.2500i   1.0000 + 1.2500i  -0.5000 - 0.2500i

Spect:
    2.2500    0.5590    0.7500    0.5590
    0.5590    0.5590    1.6008    1.2500
    0.2500    1.2500    0.2500    1.2500
    0.5590    1.2500    1.6008    0.5590

Inverse:
     0     0     1     0
     2     0     0     0
     0     3     0     0
     0     1     0     2
```

```
% Program 1

% General 2D DFT
f = double(imread('cameraman.tif'), [40 40 99 99]));

[M, N] = size(f);
F = zeros(M, N);
for u = 1: M
 for v = 1: N
  F(u,v) = 0;
  for x = 1: M
   for y = 1: N
    F(u,v)=F(u,v)+f(x,y)*exp(-2j*pi*((u-1)*(x-1)/M+(v-1)*(y-1)/N));
   end
  end
 end
end
F = F / (M*N);
spect = abs(F);
```




```
% INVERTION
fnew = zeros(1, N);
for x = 1: M
 for y = 1: N
  fnew(x,y) = 0;
  for u = 1: M
   for v = 1: N
    fnew(x,y)=fnew(x,y)+F(u,v)*exp(2j*pi*((u-1)*(x-1)/M+(v-1)*(y-1)/N));
   end
  end
 end
end
fnew = round(fnew);

f_original = f;
f2d_spect = spect;
Fshift = fftshift(f2d_spect);
f_inv = fnew;

figure
      subplot(2,2,1),imshow(uint8(f_original)),title('Original');
      subplot(2,2,2),imshow(spect),title('spectrum');
      subplot(2,2,3),imshow(Fshift),title('spectrum with shift');
      subplot(2,2,4),imshow(uint8(f_inv)),title('after inverse');
```

```
% Program 2

% Square 2D DFT
f = double(imread('cameraman.tif'), [40 40 99 99]));
[M, N] = size(f);

F = zeros(N);
for u = 1: N
 for v = 1: N
  F(u,v) = 0;
  for x = 1: N
   for y = 1: N
    F(u,v) = F(u,v)+f(x,y)*exp(-2j*pi*(((u-1)*(x-1)+(v-1)*(y-1))/N));
   end
  end
 end
end

F = F / (N);
spect = abs(F);
```




```
% INVERTION
for x = 1: N
 for y = 1: N
  fnew(x,y) = 0;
  for u = 1: N
   for v = 1: N
    fnew(x,y) = fnew(x,y)+F(u,v)*exp(2j*pi*((u-1)*(x-1)+(v-1)*(y-1))/N);
   end
  end
 end
end
fnew = round(fnew/N);

f_original = f;
f2d_spect = spect;
Fshift = fftshift(f2d_spect);
f_inv = fnew;

Figure
     subplot(2,2,1),imshow(uint8(f_original)),title('Original');
     subplot(2,2,2),imshow(uint8(spect)),title('spectrum');
     subplot(2,2,3),imshow(uint8(Fshift)),title('spectrum with shift');
     subplot(2,2,4),imshow(uint8(f_inv)),title('after inverse');
```

```
clc;
clear;
% Apply DFT (separability property) to the following points
f = [0 0 1 0;
     2 0 0 0;
     0 3 0 0;
     0 1 0 2];

[M, N] = size(f);
F = zeros(M, N);
for x = 1:M
    for v = 1:N
        F(x, v) = 0;
        for y = 1:N
            F(x, v) = F(x, v)+ f(x, y) * exp(-2j * pi  * (v-1) * (y-1)/N);
        end
    end
end
F2 = zeros(M, N);
for u = 1:M
    for v = 1:N
        F2(u, v) = 0;
        for x = 1:N
            F2(u, v) = F2(u, v) + F(x, v) * exp(-2j * pi * (u-1) * (x-1)/N);
        end
    end
end
F2 = F2 / N;
disp('DFT:');
disp(F2);

spect = abs(F2);
disp('Fspect:');
disp(spect);

fnew = zeros(M, N);
for u = 1:N
    for y = 1:N
        fnew(u, y) = 0;
        for v = 1:N
            fnew(u, y) = fnew(u, y) +  F2(u, v) * exp(2j * pi * (v-1)*(y-1)/N);
        end
    end
end
fnew2 = zeros(M, N);
for x = 1:N
    for y = 1:N
        fnew2(x, y) = 0;
        for u = 1:N
            fnew2(x, y) = fnew2(x, y)+ fnew(u, y) * exp(2j * pi * (u-1)*(x-1)/N);
        end
    end
end
fnew2 = round(fnew2/N);
disp('Inverse: ');
disp(fnew2)
```

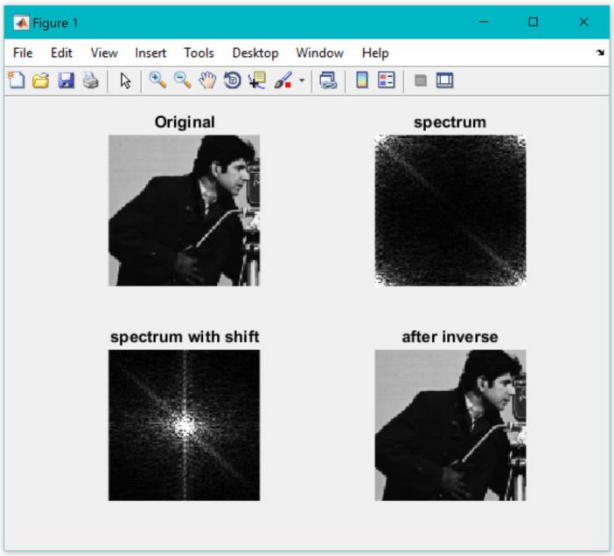
Command Window

```
DFT separability:
   2.2500 + 0.0000i   0.2500 - 0.5000i  -0.7500 - 0.0000i   0.2500 + 0.5000i
  -0.5000 + 0.2500i  -0.5000 + 0.2500i   1.0000 - 1.2500i   0.0000 - 1.2500i
  -0.2500 - 0.0000i  -0.7500 - 1.0000i  -0.2500 + 0.0000i  -0.7500 + 1.0000i
  -0.5000 - 0.2500i  -0.0000 + 1.2500i   1.0000 + 1.2500i  -0.5000 - 0.2500i

Fspect:
    2.2500    0.5590    0.7500    0.5590
    0.5590    0.5590    1.6008    1.2500
    0.2500    1.2500    0.2500    1.2500
    0.5590    1.2500    1.6008    0.5590

Inverse:
     0     0     1     0
     2     0     0     0
     0     3     0     0
     0     1     0     2
```

Lab 8

```
clc;
clear;
% program
% Apply 1D DFT to 2D image without transforming it into 1D image.
f = double(imcrop(imread('cameraman.tif'),[40 40 99 99]));
[M, N] = size(f);
F = zeros(M, N);
for x = 1:M
    for v = 1:N
        F(x, v) = 0;
        for y = 1:N
            F(x, v) = F(x, v)+ f(x, y) * exp(-2j * pi  * (v-1) * (y-1)/N);
        end
    end
end
F2 = zeros(M, N);
for u = 1:M
    for v = 1:N
        F2(u, v) = 0;
        for x = 1:N
            F2(u, v) = F2(u, v) + F(x, v) * exp(-2j * pi * (u-1) * (x-1)/N);
        end
    end
end
F2 = F2 / N;
spect = abs(F2);

fnew = zeros(M, N);
for u = 1:N
    for y = 1:N
        fnew(u, y) = 0;
        for v = 1:N
            fnew(u, y) = fnew(u, y) +  F2(u, v) * exp(2j * pi * (v-1)*(y-1)/N);
        end
    end
end
```

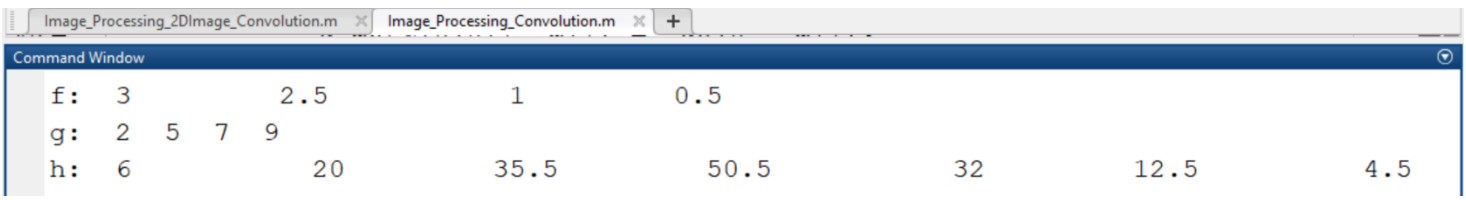
```
fnew2 = zeros(M, N);
for x = 1:N
    for y = 1:N
        fnew2(x, y) = 0;
        for u = 1:N
            fnew2(x, y) = fnew2(x, y)+ fnew(u, y) * exp(2j * pi * (u-1)*(x-1)/N);
        end
    end
end


fnew2 = round(fnew2/N);

f_original = f;
f2d_spect = spect;
Fshift = fftshift(f2d_spect);
f_inv = fnew2;
```




```
figure
      subplot(2,2,1),imshow(uint8(f_original)),title('Original');
      subplot(2,2,2),imshow(uint8(f2d_spect)),title('spectrum');
      subplot(2,2,3),imshow(uint8(Fshift)),title('spectrum with shift');
      subplot(2,2,4),imshow(uint8(f_inv)),title('after inverse');
```

Lab 9

```
% CONVOLUTION
% program 1 using equation
f = [3, 2.5, 1, 0.5];
g = [2, 5, 7, 9];
A = length(f);
B = length(g);
M = A + B - 1;
fe = [f, zeros(1, M - A)];
ge = [g, zeros(1, M - B)];
h = zeros(1, M);
for m = 0:M-1
    for x = 0:M-1
        if (x - m+1) <= 0
            h(x + 1) = h(x+1) + fe(m+1) * ge(M + (x - m+1));
        else
            h(x + 1) = h(x+1) + fe(m+1) * ge(x - m+1);
        end
    end
end
disp(['f:   ', num2str(f)]);
disp(['g:   ', num2str(g)]);
disp(['h:   ', num2str(h)]);
```



```
disp('----------------------------------');
disp('------program 2 using Matrix------');
disp('----------------------------------');
f_matrix = fe;
g_matrix = zeros(M, M);
h = zeros(1, M);
for m = 0:M-1
    for x = 0:M-1
        if (x - m+1) <= 0
            g_matrix(x+1, m+1) = ge(M + (x - m+1));
        else
            g_matrix(x+1, m+1) =  ge(x - m+1);
        end
    end
end
disp('g_matrix:');
disp(g_matrix);
for x=1:M
    h(x) = sum(f_matrix.* g_matrix(x,:));
end
disp(['h: ', num2str(h)]);
```

```
----------------------------------
------program 2 using Matrix------
----------------------------------
g_matrix:
     2     0     0     0     9     7     5
     5     2     0     0     0     9     7
     7     5     2     0     0     0     9
     9     7     5     2     0     0     0
     0     9     7     5     2     0     0
     0     0     9     7     5     2     0
     0     0     0     9     7     5     2

h: 6          20        35.5        50.5          32        12.5         4.5
```

Lab 10

```
clc;
clear;
% CORRELATION
% program 1 using equation
f = [3, 2.5, 1, 0.5];
g = [2, 5, 7, 9];
A = length(f);
B = length(g);
M = A + B - 1;
fe = [f, zeros(1, M - A)];
ge = [g, zeros(1, M - B)];
h = zeros(1, M);
for m = 0:M-1
    for x = 0:M-1
        if (x + (m+1)) > M
            h(x + 1) = h(x+1) + fe(m+1) * ge((x + (m)) - (M-1));
        else
            h(x + 1) = h(x+1) + fe(m+1) * ge(x + (m+1));
        end
    end
end
```

```
disp('---------------------------------');
disp('Correlation using Equation:');
disp('---------------------------------');

disp(['f: ', num2str(f)]);
disp(['g: ', num2str(g)]);
disp(['h: ', num2str(h)]);


disp('---------------------------------');
disp('------program 2 using Matrix------');
disp('---------------------------------');

f_matrix = fe;
g_matrix = zeros(M, M);


h = zeros(1, M);

for m = 0:M-1
    for x = 0:M-1
        if (x + (m+1)) <= M
            g_matrix(m+1, x+1) =  ge(x + m+1);
        else
            g_matrix(m+1, x+1) = ge((x + m) - (M-1));
        end
    end
end
disp('g_matrix:')
disp(g_matrix)

for i=0:M-1
    h(i+1) = sum(f_matrix.* g_matrix(i+1,:));
end
disp(['h: ', num2str(h)]);
```

```
---------------------------------
Correlation using Equation:
---------------------------------
f: 3         2.5           1         0.5
g: 2  5  7  9
h: 30          41.5          43.5           27             1           4.5          13.5
---------------------------------
------program 2 using Matrix------
---------------------------------
g_matrix:
     2     5     7     9     0     0     0
     5     7     9     0     0     0     2
     7     9     0     0     0     2     5
     9     0     0     0     2     5     7
     0     0     0     2     5     7     9
     0     0     2     5     7     9     0
     0     2     5     7     9     0     0

h: 30          41.5          43.5           27             1           4.5          13.5
```