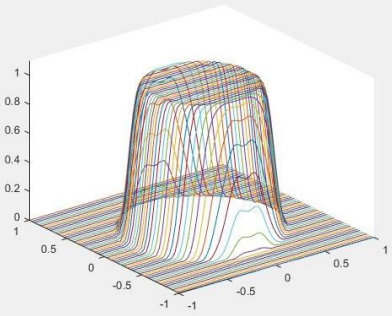
```
%                          DSP_Project_2
%
% Design of Two-Dimensional FIR Linear-phase Filters
%
clear all;   % clear workspace
clc;         % clear command window
%
% circularly symmetric low-pass filter
%
N = 25;
wp = 0.15*pi;
ws = 0.35*pi;
NS = 50;
%
%
% commonly use parameters
NH = (N-1)/2; % (25-1)/2 = 12
NH1 = NH + 1; % 12+1 = 13
NH2 = (NH1)^2; % 所有係數個數 13^2 = 169
deltaw = pi/NS;
%
%
PS = 0; % pass-band sampling points(取樣總點數)
SS = 0; % stop-band sampling points
for s1=0:NS
    w1 = s1*deltaw;
    for s2=0:NS
        w2 = s2*deltaw;
        if (w1^2 + w2^2)^0.5 <= wp
            PS = PS + 1;
        elseif (w1^2 + w2^2)^0.5 >= ws
            SS = SS + 1;
        end
    end
end
%
%
P = zeros(NH2,1);
Qp = zeros(NH2,NH2);
Qs = zeros(NH2, NH2);
for i=0:NH2-1
    i1 = mod(i,NH1); % mod 取餘數
    i2 = floor(i/NH1); % floor 下取整函數
                       % ceil 上取整函數
    for s1=0:NS
    w1 = s1*deltaw;
```

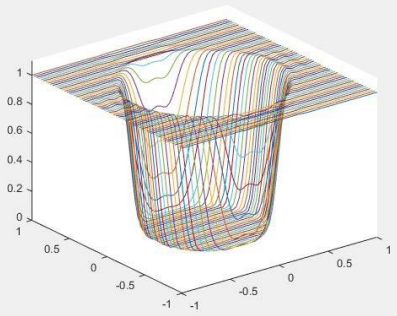
```
        for s2=0:NS
            w2 = s2*deltaw;
            if (w1^2 + w2^2)^0.5 <= wp
                P(i+1) = P(i+1) + cos(i1*w1)*cos(i2*w2);
            end
        end
    end
    for j=0:NH2-1
        j1 = mod(j,NH1);
        j2 = floor(j/NH1);
        for s1=0:NS
        w1 = s1*deltaw;
            for s2=0:NS
                w2 = s2*deltaw;
                if (w1^2 + w2^2)^0.5 <= wp
                    Qp(i+1,j+1) = Qp(i+1,j+1) +
cos(i1*w1)*cos(i2*w2)*cos(j1*w1)*cos(j2*w2);
                elseif (w1^2 + w2^2)^0.5 >= ws
                    Qs(i+1,j+1) = Qs(i+1,j+1) +
cos(i1*w1)*cos(i2*w2)*cos(j1*w1)*cos(j2*w2);
                end
            end
        end

    end
end
P = 0.25*wp^2*pi*(-2)*P/PS;
Qp = 0.25*wp^2*pi*Qp/PS;
Qs = (pi^2-0.25*ws^2*pi)*Qs/SS;
Q = Qp + Qs;
A = -0.5*inv(Q)*P;
%
%
A = reshape(A,NH1,NH1);
h = zeros(N,N);
h(NH+1,NH+1) = A(1,1);
h(1:NH,NH+1) = 0.5*A(NH+1:-1:2,1);
h(NH+2:N,NH+1) = 0.5*A(2:NH+1,1); % A的2~NH+1放在第一個column
%
h(NH+1,1:NH) = 0.5*A(1,NH+1:-1:2);
h(NH+1,NH+2:N) = 0.5*A(1,2:NH+1);
%
h(1:NH,1:NH) = 0.25*A(NH+1:-1:2,NH+1:-1:2);
h(NH+2:N,1:NH) = 0.25*A(2:NH+1,NH+1:-1:2);
h(1:NH,NH+2:N) = 0.25*A(NH+1:-1:2,2:NH+1);
h(NH+2:N,NH+2:N) = 0.25*A(2:NH+1,2:NH+1);
hL = h;
```

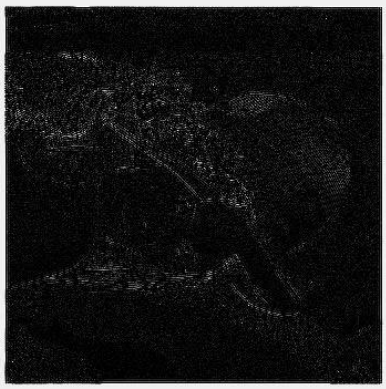
```
%
MR = abs(freqz2(h,-1:2/64:1,-1:2/64:1));
XX = zeros(65,65);
YY = zeros(65,65);
for i=1:65
   XX(:,i) = (-1:2/64:1)';
   YY(i,:) = -1:2/64:1;
end
subplot(2,3,2);
plot3(XX,YY,MR);
axis([-1,1,-1,1,0,1.1]);
% xlabel('Normalized frequency(\omega_1/\pi)');
% ylabel('Normalized frequency(\omega_2/\pi)');
% zlabel('Magnitude Response');
%
%
% circularly symmetric highpass filter
%
temp = wp;
wp = ws;
ws = temp;
%
%
%
PS = 0; % pass-band sampling points
SS = 0; % stop-band sampling points
for s1=0:NS
    w1 = s1*deltaw;
    for s2=0:NS
        w2 = s2*deltaw;
        if (w1^2 + w2^2)^0.5 >= wp
            PS = PS + 1;
        elseif (w1^2 + w2^2)^0.5 <= ws
            SS = SS + 1;
        end
    end
end
%
%
P = zeros(NH2,1);
Qp = zeros(NH2,NH2);
Qs = zeros(NH2, NH2);
for i=0:NH2-1
    i1 = mod(i,NH1);
    i2 = floor(i/NH1);
    for s1=0:NS
    w1 = s1*deltaw;
```

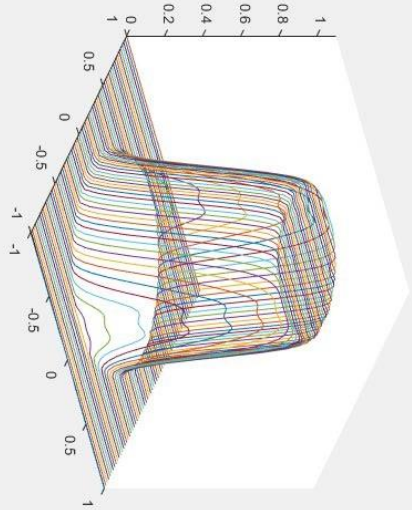
```
        for s2=0:NS
            w2 = s2*deltaw;
            if (w1^2 + w2^2)^0.5 >= wp
                P(i+1) = P(i+1) + cos(i1*w1)*cos(i2*w2);
            end
        end
    end
    for j=0:NH2-1
        j1 = mod(j,NH1);
        j2 = floor(j/NH1);
        for s1=0:NS
        w1 = s1*deltaw;
            for s2=0:NS
                w2 = s2*deltaw;
                if (w1^2 + w2^2)^0.5 >= wp
                    Qp(i+1,j+1) = Qp(i+1,j+1) +
cos(i1*w1)*cos(i2*w2)*cos(j1*w1)*cos(j2*w2);
                elseif (w1^2 + w2^2)^0.5 <= ws
                    Qs(i+1,j+1) = Qs(i+1,j+1) +
cos(i1*w1)*cos(i2*w2)*cos(j1*w1)*cos(j2*w2);
                end
            end
        end
    end
end
P = 0.25*wp^2*pi*(-2)*P/PS;
Qp = 0.25*wp^2*pi*Qp/PS;
Qs = (pi^2-0.25*ws^2*pi)*Qs/SS;
Q = Qp + Qs;
A = -0.5*inv(Q)*P;
%
%
A = reshape(A,NH1,NH1);
h = zeros(N,N);
h(NH+1,NH+1) = A(1,1);
h(1:NH,NH+1) = 0.5*A(NH+1:-1:2,1);
h(NH+2:N,NH+1) = 0.5*A(2:NH+1,1);
%
h(NH+1,1:NH) = 0.5*A(1,NH+1:-1:2);
h(NH+1,NH+2:N) = 0.5*A(1,2:NH+1);
%
h(1:NH,1:NH) = 0.25*A(NH+1:-1:2,NH+1:-1:2);
h(NH+2:N,1:NH) = 0.25*A(2:NH+1,NH+1:-1:2);
h(1:NH,NH+2:N) = 0.25*A(NH+1:-1:2,2:NH+1);
h(NH+2:N,NH+2:N) = 0.25*A(2:NH+1,2:NH+1);
hH = h;
%
```

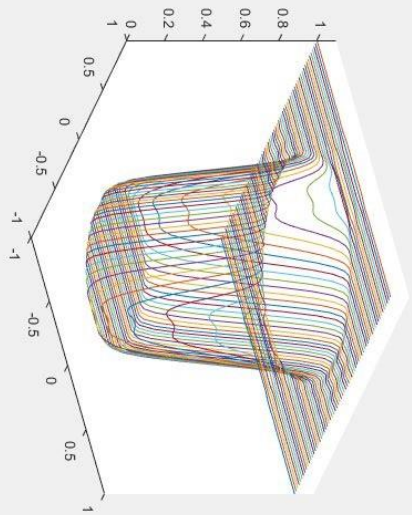
```
MR = abs(freqz2(h,-1:2/64:1,-1:2/64:1));
XX = zeros(65,65);
YY = zeros(65,65);
for i=1:65
   XX(:,i) = (-1:2/64:1)';
   YY(i,:) = -1:2/64:1;
end
subplot(2,3,3);
plot3(XX,YY,MR);
axis([-1,1,-1,1,0,1.1]);
% xlabel('Normalized frequency(\omega_1/\pi)');
% ylabel('Normalized frequency(\omega_2/\pi)');
% zlabel('Magnitude Response');
%
% Image simulation
%
lena = imread('lena.bmp');
subplot(2,3,4);
imshow(uint8(lena)); % unsigned-8bit(256x256)
lena_L = filter2(hL,lena);
subplot(2,3,5);
imshow(uint8(lena_L));
lena_H = 10*filter2(hH,lena);
subplot(2,3,6);
imshow(uint8(lena_H));
%
% Two-Dimensional FIR Linear-phase Filter
```








0
0.2
0.4
0.6
0.8
1
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1


0
0.2
0.4
0.6
0.8
1
1
0.5
0
-0.5
-1
-1
-0.5
0
0.5
1