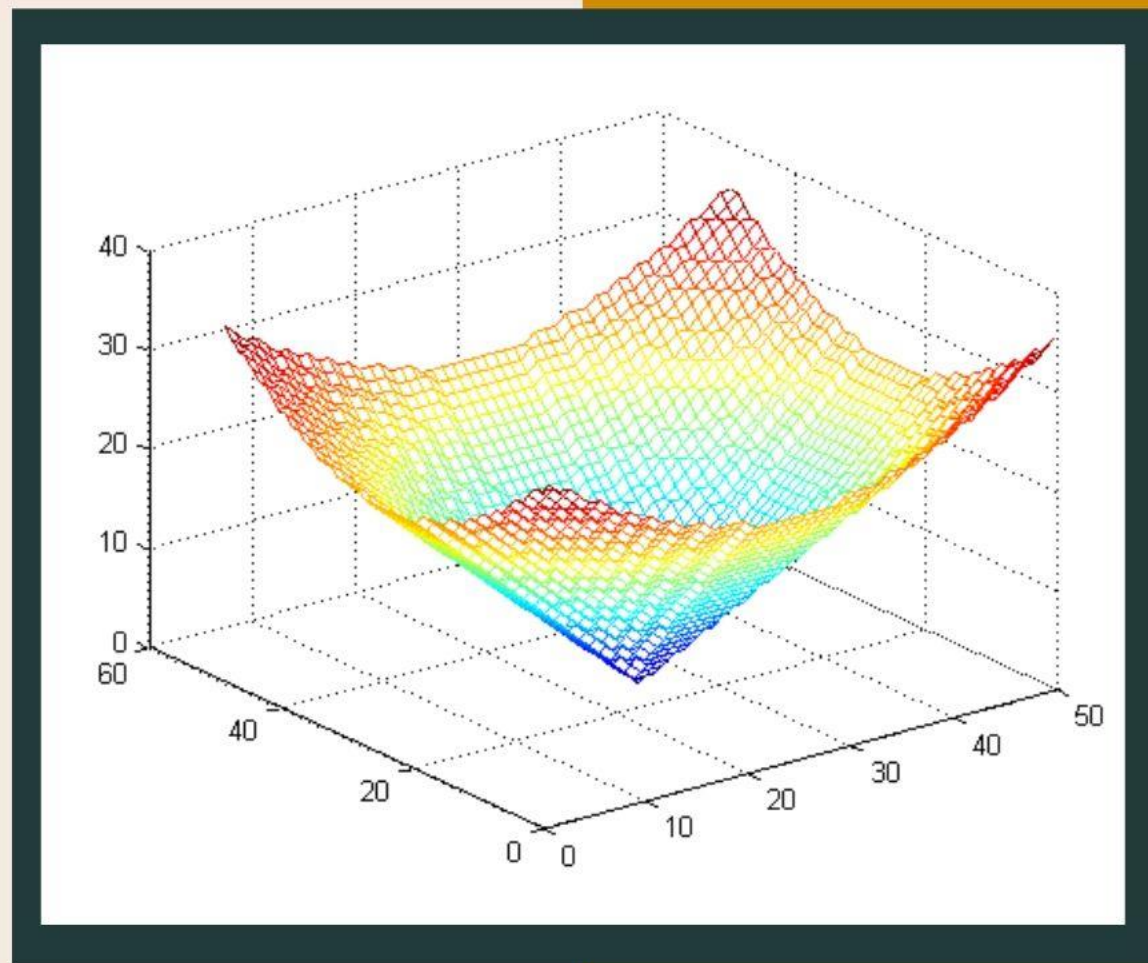


LAB:04

DISTANCE / RESIZE

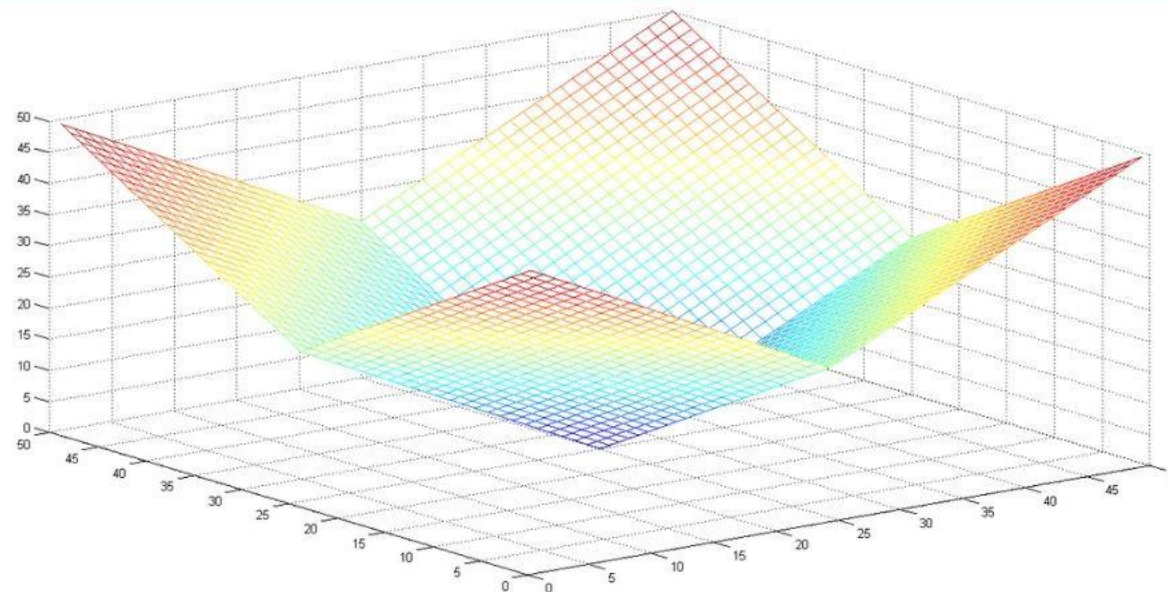


1.) กำหนดให้มี image ขนาด 50x50 ให้เก็บค่าของ Euclidean Distance โดยให้ center ของ image มีระยะทางเป็น 0 แล้ว plot ค่าออกมาเป็นสามมิติ โดยใช้คำสั่ง mesh

```
clc,clear,close all;  
output = zeros(50,50);  
[row,col] = size(output);  
for y=1:row  
    for x=1:col  
        output(y,x) = sqrt(((25-x)^2)+((25-y)^2));  
    end  
end  
int_output = uint8(output);  
m_output(1:50,1:50) = 0;  
m_output(1:50,1:50) = int_output(1:50,1:50);  
imshow(int_output);  
figure;  
mesh(m_output);
```

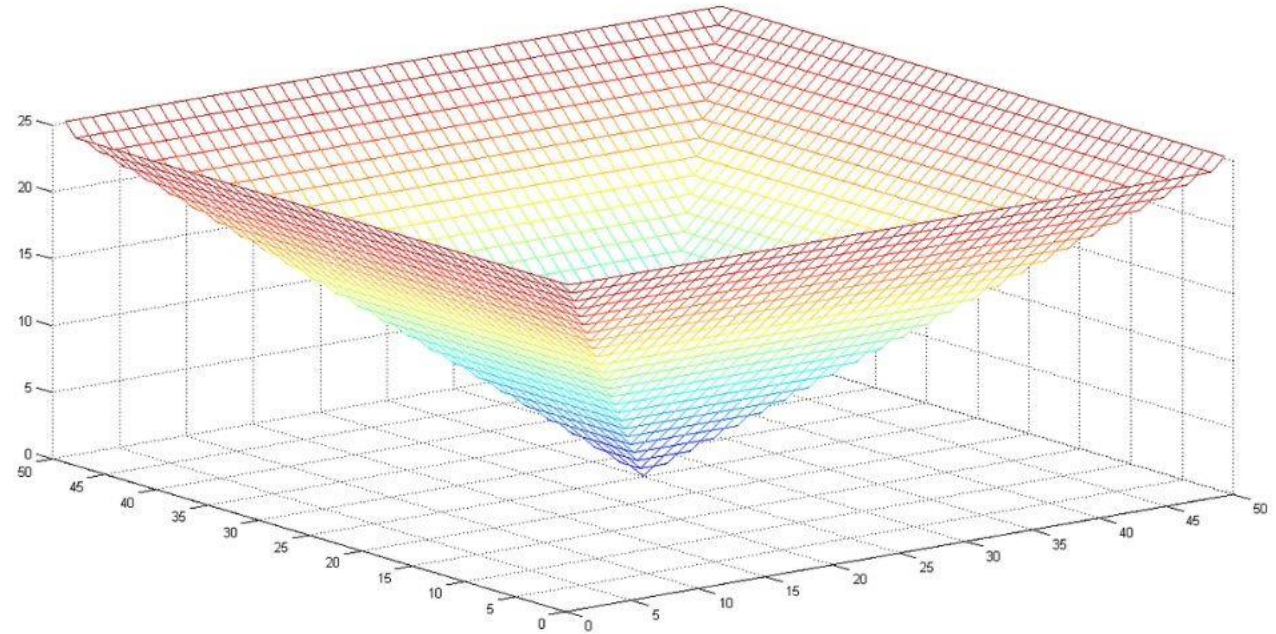
2.) กำหนดให้มี image ขนาด 50x50 ให้เก็บค่าของ City-Block Distance โดยให้ center ของ image มีระยะทางเป็น 0 แล้ว plot ค่าออกมาเป็นสามมิติ โดยใช้คำสั่ง mesh

```
clc,clear,close all;  
img = uint8(zeros(50,50));  
img(25,25) = 1 ;  
D = bwdist(img,'cityblock');  
figure;  
imcontour(D), title('City-block');  
img2(1:50,1:50) = 0;  
img2(1:50,1:50) = D(1:50,1:50);  
figure;  
mesh(img2);
```



3.) กำหนดให้มี image ขนาด 50x50 ให้เก็บค่าของ Chessboard Distance โดยให้ center ของ image มีระยะทางเป็น 0 แล้ว plot ค่าออกมาเป็นสามมิติ โดยใช้คำสั่ง mesh

```
clc, clear, close all
imgC = zeros(50,50);
imgC(25,25) = 0;
[ny,nx] = size(imgC);
i = 1;
j = 1;
for i = 1:ny
    for j = 1:nx
        if i <= 50 && j <= 50
            imgC(i,j) = max(abs(25-i),abs(25-j));
        end
        j = j+1;
    end
    i = i+1;
end
mesh(imgC)
```



4.) ให้เขียนโปรแกรมเพื่อลดหรือขยายภาพโดยใช้หลักการของ Nearest Neighbor Interpolation แล้วเปรียบเทียบกับการใช้ฟังก์ชัน function “imresize” โดยเริ่มจากขยายภาพจาก 4x4 เป็น 6x10 แล้วคำนวณด้วยมือก่อน

```
clc,clear,close all;
```

```
%ประกาศค่ารูปภาพเริ่มต้นเริ่มต้น
```

```
img_Before = [ 25 50 25 50 ; 50 100 150 200 ; 25 50 25 50 ; 200 150  
100 50];
```

```
%กำหนดขนาดเป้าหมาย
```

```
sizeRow = 6;
```

```
sizeColumn = 10;
```

```
img_After = uint8(zeros(sizeRow,sizeColumn));
```

```
%คำนวณค่า
```

```
cal_Row = (4-1)/(sizeRow-1);
```

```
cal_Column = (4-1)/(sizeColumn-1);
```

```
min_y = 1;
```

```
max_y = 2;
```

```
add_value_y = 1;
```

```
for i=1:sizeRow
```

```
    min_x=1;
```

```
    max_x=2;
```

```
    add_value_x = 1;
```

```
    if add_value_y >= max_y
```

```
        min_y = min_y+1;
```

```
        max_y = max_y+1;
```

```
    end
```

```
    for j=1:sizeColumn
```

```
        if add_value_x >= max_x
```

```
            min_x = min_x+1;
```

```
            max_x = max_x+1;
```

```
        end
```

```
        x = min_x;
```

```
        y = min_y;
```

```
        if min_x+0.5 < add_value_x
```

```
            x = max_x;
```

```
        end
```

```
        if min_y+0.5 < add_value_y
```

```
            y = max_y;
```

```
        end
```

```
        img_After(i,j) = img_Before(y,x);
```

```
        add_value_x = add_value_x + cal_Column;
```

```
        end
```

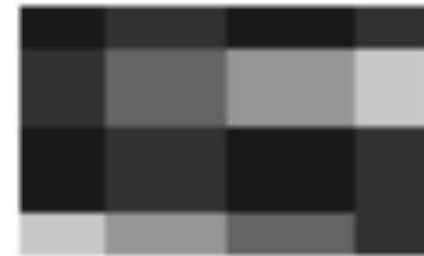
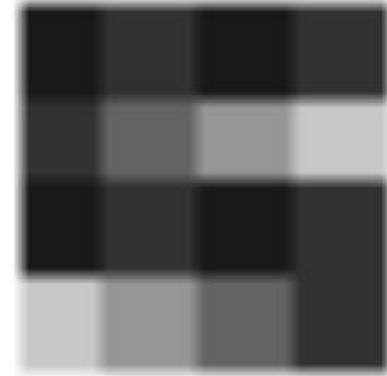
```
        add_value_y = add_value_y+cal_Row;
```

```
    end
```

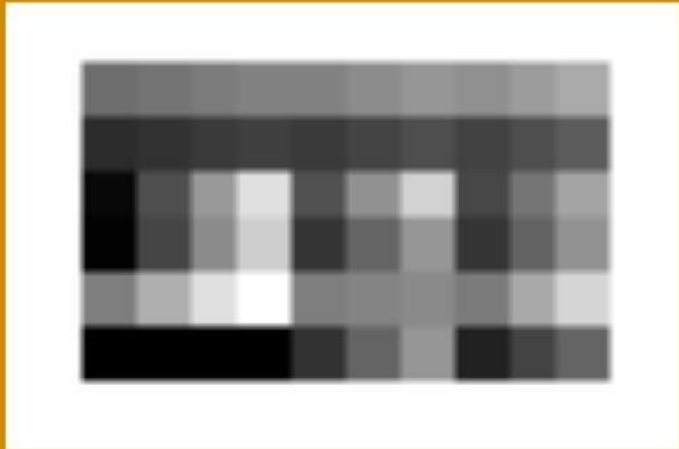
```
    imshow(uint8(img_Before));
```

```
    figure
```

```
    imshow(uint8(img_After));
```



5.) ให้เขียนโปรแกรมเพื่อลดหรือขยายภาพโดยใช้หลักการของ bilinear Interpolation แล้วเปรียบเทียบกับการใช้ฟังก์ชัน function “imresize” โดยเริ่มจากขยายภาพจาก 4x4 เป็น 6x10 แล้วคำนวณด้วยมือก่อน



```
clc,clear,close all;
img = uint8(zeros(4,4));
reimg = uint8(zeros(6,10));
[row,col] = size(reimg);
img(1:4,1:4) = [110 120 130 140;10 20 30 40;210 200 130
140;255 0 150 100];
avg_y = (4-1)/(row-1);
avg_x = (4-1)/(col-1);
int_min_y = 1;
int_max_y = 2;
add_value_y = 1;
for y = 1:row;
    int_min_x = 1;
    int_max_x = 2;
    add_value_x = 1;
    if int_max_y < add_value_y
        int_max_y = int_max_y+1;
        int_min_y = int_min_y+1;
    end
    for x = 1:col;
        if int_max_x < add_value_x
            int_min_x = int_min_x+1;
            int_max_x = int_max_x+1;
        end
```

```
t = add_value_x - int_min_x;
u = add_value_y - int_min_y;
reimg(y,x) = ((1-t)*(1-u)*img(int_min_y,int_min_x))+
((t)*(1-u)*img(int_min_y,int_max_x))+((t)*(1-u)*img(int_max_y,int_min_x))+((t)*
(u)*img(int_max_y,int_max_x));
add_value_x = add_value_x + avg_x;
end
add_value_y = add_value_y + avg_y;
end
imshow(reimg);
```

THANK YOU