# 電腦視覺（一）Homework6

# R11521701  程懷恩

## 1. Yokoi Connectivity Number

```
11111111        12111111111122322221     111111111111      0  0
15555551        115555555511 2 11  11   1155555555511       0
15555551      1 2115555112  21112221    155555555551      21
15555551      1 2 155112 22221511        155555555511      1
15555551        22 2112 22    121 0 0    15555555555511    0
15555551       1  2  21 2     1   1      155555555551  0
15555551         12 1  121111    1321    155555555555511
15111551          1322 1155551111        155555555555551
111 1551          1  121555555511        155555555555511
11  1551             21155555511         15511155555511
21  1551             2 15555555111       1551 11555511
1   1551            2 15555555511    1551  115551           1
    1551           1121155555555551   1551   15511         12
    1551           15555555555555511  1551    1111        111
    1551      1     2221155555555555511 1151     11       1151
    1551      2    22 1 155555555555511 151  11111       1551
    1551      2    1   1155555555555551 151 115551       11551
    1551      2        1155555555555555111511155511      115551
    1551      12       115555555555555555555555555551      155551
    1551      11   0 2215555555555555555555555555112       1155551
    1551      111   22 15555555555555555555555555551 1      1555551
    1551      1511  1 1251121111121115555555555111       11555551
    1551      15521  1 121 1 11  1  155555555111  0       15555551
    1551      1151  132 2        1155555111   0          115555551
    1551      151 0  322         115555111  121          155555551
    1551      1221    2           155551   131          1155555551
    1551       2  0  1            115555511   1          1155555551
    1551       2   0      0    1155555551  0      1 155555551
    1551       2           115555555551            21155555551
    1551       1  0           115555555551            15555555551
    1551          1          11511111555521  1       1155555555551
    1551        1 1         11111  1155511   2       155555555551
    1551        131          111    15111   2       155555555551
    1551      121 0        1121   1  111  1  2       1155555555551
    1551       11          111 1  221 11  1  2       1555555555551
    1551      12  0    1      21 121  11 1111   2      15555555555551
    1551      1      12      22 151111111551   2     1155555555551
    1551    1          2    1555551115511   1    155555555551
    1551    2    0      0 22  12555551 15551    1  1555555555551
    1551    1              1    155551 11511    2 11555555555551
    1551       0 0         21     155551 1 151   2 155555555555551
    1551               2      15555112 151    2 155555555555551
    1551          1  1 1     1155555511111    2 155555555555551
    1551          2  22     111511111212     21155555555555551
    1551 0        1 12        151    2 1     155555555111555551
    1551       0  0  0        1111   121     155555551 1555551
    1551            0           11111111     155555551 1555551
    1551       0              115551       155555551 1555511
    1551                       15551        211111111 155511
    11521     1   12          122155511      2    11 115511
1    151 0    1   1          155555111    2111      15511
22   1511        1           15555555111  155111    1511
 22  1511        1           155555555551  155551  1151
  2  151        0 1         11155555555511  155511  1511
  2  1521  0       1         155555555555511 15551 12151
  2  151         121         155555555555551 155511 1551
  2  1511              0  155555555555551 115551 1511
 21  1511          11       155555555555551  1111111151
 11  151      0            1155555555555555511   111511
 11  151                  15555555555555551      151
 11  151        0        115555555555555551     211
 11  151                 11555555555555555511    1
 11  151               0 155555555555555551
 11  111        0        12111111111111111111
```

將輸出的結果寫入 txt 後截圖

```python
def DownSampling(img):
    arr = np.zeros((img.shape[0]//8+2, img.shape[1]//8+2),np.uint8)
    for i in range(1, arr.shape[0]-1):
        for j in range(1, arr.shape[1]-1):
            arr[i][j]=img[(i-1)*8][(j-1)*8]
    return arr

def Yokoi(img,i,j):
    if(img[i][j]==0):return ' '
    if(img[i-1][j-1]!=0 and img[i-1][j]!=0 and img[i-1][j+1]!=0
       and img[i][j-1]!=0 and img[i][j+1]!=0
       and img[i+1][j-1]!=0 and img[i+1][j]!=0 and img[i+1][j+1]!=0):
        return '5'
    count=0
    #right
    if(img[i][j+1]!=0):
        if(img[i-1][j]==0 or img[i-1][j+1]==0):count+=1
    #top
    if(img[i-1][j]!=0):
        if(img[i-1][j-1]==0 or img[i][j-1]==0):count+=1
    #left
    if(img[i][j-1]!=0):
        if(img[i+1][j-1]==0 or img[i+1][j]==0):count+=1
    #bottom
    if(img[i+1][j]!=0):
        if(img[i+1][j+1]==0 or img[i][j+1]==0):count+=1

    if(count==0):return '0'
    return str(count)
```

Coding:

首先 downsampling 時，將原結果之 64*64 矩陣，行列頭尾各加入一層

空陣列，使得結果變為 66*66 之陣列，省去考慮邊界條件。

Yokoi 的函示將 h(b,c,d,e)求得 a1,a2,a3,a4 之結果以 count 變數紀錄，最

後根據該格計算總數輸出對應編碼。