



COMPUTER VISION 1

Homework 6

姓名：蘇宛琳

系所：電信所碩一

學號：R05942060

指導教授：傅楸善老師

Computer Vision Report – Homework 6

R05942060 蘇宛琳

Question :

Write a program to generate Yokoi connectivity number. Down sample lena.bmp from 512*512 to 64*64 first. Sample pixels positions at each 8*8 top-left corner, so everyone will get the same answer .



lena.bmp



DownsampleLENA.bmp



binarizelena.bmp

用 h function 比較中心點 x_0 與四個角 x_1, x_2, x_3, x_4 後得到四個值 a_1, a_2, a_3, a_4 。
最後在利用 f function 對這四個值 a_1, a_2, a_3, a_4 做運算，得到該中心點 x_0 的 Yokoi Connectivity Number 。

Example.

$$a_1 = h(x_0, x_1, x_6, x_2)$$

$$a_2 = h(x_0, x_2, x_7, x_3)$$

$$a_3 = h(x_0, x_3, x_8, x_4)$$

$$a_4 = h(x_0, x_4, x_5, x_1)$$

$$y = f(a_1, a_2, a_3, a_4)$$

	1	1
1	1	1
	1	

$$a_1 = r$$

$$a_2 = q$$

$$a_3 = q$$

$$a_4 = q$$

$$f(a_1, a_2, a_3, a_4) = 3$$

(branching)

* Yokoi Connectivity Number Concept*

Step1. 先將灰階的 512×512 每 8 點取 1 點降取到 64×64 ，並以 128 為門檻值做二元化。

Step2. 利用 hfunction 比較中心點 x_0 與四個角 x_1, x_2, x_3, x_4 後得到四個值 a_1, a_2, a_3, a_4 。

Step3. 利用 f function 對這四個值 a_1, a_2, a_3, a_4 做運算，得到該中心點 x_0 的 Yokoi Connectivity Number。

Source code (Main code)

```
clear;
```

```
close;
```

Grayscale LENA image

```
LENA = imread('lena.bmp');
```

```
INFO = imfinfo('lena.bmp');
```

Down Sample from 512x512 to 64x64

```
for x = 1 : INFO.Height/8,  
    for y = 1 : INFO.Width/8,  
        NEWLENA(x,y) = LENA(x*8-7,y*8-7);  
    end;  
end;
```

Sample pixels positions at each 8*8 top-left corner

```
imwrite(NEWLENA, 'DownsampleLENA.bmp')
```

Binarize LENA image

```
for x = 1 : INFO.Height/8,  
    for y = 1 : INFO.Width/8,
```

```

T = 128;

if NEWLENA(x,y) > T,
    NEWLENA(x,y) = 255;
else
    NEWLENA(x,y) = 0;
end;
end;
end;

imwrite(NEWLENA, 'binarizelena.bmp')

```

Call Yokoi Connectivity function

```
output = YokoiConnectivity(NEWLENA);
```

Print Result Label & Write Result to txt file

```

fid = fopen('Yokoi Connectivity Number.txt','w');

for k = 1 : size(output,1)

    fprintf(fid,'%c',output(k,:));

    fprintf(fid,'\r\n');

end

fclose(fid);

```



lena.bmp



DownsampleLENA.bmp



binarizelena.bmp

* Yokoi Connectivity function Concept*

先將灰階的 512*512 每 8 點取 1 點降取到 64*64，並以 128 為門檻值做二元化。

利用 hfunction 比較中心點 x0 與四個角 x1, x2, x3, x4 後得到四個值 a1, a2, a3, a4。

利用 f function 對這四個值 a1, a2, a3, a4 做運算，得到該中心點 x0 的 Yokoi Connectivity Number。

Connectivity number (0~5) 把 border 的情況分類成 6 大類：

0	Isolated	(周圍都屬背景，沒有鄰居)
1	Edge	(周圍1個鄰居)
2	Connecting	(周圍2個鄰居，恰在線上)
3	Branching	(周圍3個鄰居)
4	Crossing	(周圍4個鄰居)
5	Interior	

```
function output = YokoiConnectivity(NEWLENA)
```

```
Ib = zeros(size(NEWLENA,1)+2,size(NEWLENA,1)+2);
```

```
Ib(2:end-1,2:end-1) = NEWLENA;
```

```
[r,c] = find(Ib);
```

```
output = char(size(NEWLENA));
```

```
for i = 1 : length(r)
```

```
    mask=Ib(r(i)-1:r(i)+1,c(i)-1:c(i)+1);
```

```
    a=zeros(1,4);
```

```
    a(1)=h(mask(5),mask(8),mask(7),mask(4)); % x0,x1,x6,x2
```

```
    a(2)=h(mask(5),mask(4),mask(1),mask(2)); % x0,x2,x7,x3
```

```
    a(3)=h(mask(5),mask(2),mask(3),mask(6)); % x0,x3,x8,x4
```

```
    a(4)=h(mask(5),mask(6),mask(9),mask(8)); % x0,x4,x5,x1
```

```
    output(r(i)-1,c(i)-1)=f(a);
```

```
end
```

```
end
```

* h function Concept*

$$h(x_0, c, d, e) = \begin{cases} q & x_0 = c; x_0 \neq d \text{ or } x_0 \neq e & (x_0 \text{ 是在邊界}) \\ r & x_0 = c = d = e & (x_0 \text{ 是在內部}) \\ s & x_0 \neq c & (\text{沒甚麼意義}) \end{cases}$$

```
function output = h(b,c,d,e)
```

```
if b == c,
    if d == b && e == b,
        output = 'r';
    else
        output = 'q';
    end
else
    output = 's';
end
end
```

* f function Concept*

$$f(a_1, a_2, a_3, a_4) = \begin{cases} 5 & \text{if } a_1 = a_2 = a_3 = a_4 = r \\ n & n = \#\{a_k \mid a_k = q\} \end{cases} \quad (\text{如此便知道border的種類})$$

```
function n = f(a)
```

```
if all(a == 'r')
    n = num2str(5);
else
    n=num2str(sum(a == 'q'));
end
end
```

* 直接貼過來的結果數據 *

11111111	121	1111111122322221	111111111111	0
15555551	121	15555511 2 11 1	115555555511	
15555551	1	2221555112 2111222	155555555551	0
15555551	1 2	155112 1 11511	1555555555511	0
15555551	1	2112 1 121 0	15555555555511	
15555551	1	21 1 1	11555555555551	
15555551	1	1 121111 11 0	155555555555511	
15111551	1	12 1255551111	115555555555551	
111 1551	0 0	1555555511	155555555555511	
11 1551		21155555511	15511155555511	
21 1551		2 15555555111	1551 11555511	
1 1551		2 1555555555511	1551 115551	1
1551		1121155555555551	1551 15511	12
1551		15111555555555511	1151 1111	111
1551	1	2221 155555555555511	151 11	1151
1551	2	12 1 11555555555555511	151 11111	1551
1551	2	11555555555555551	151 115551	11551
1551	2	11555555555555555111511155511		115551
1551	12	115555555555555555555555555551		155551
1551	11	0 22155555555555555555555555555112		1155551
1551	111	22 155555555555555555555555555551	1	1555551
1551	1511	1 125112111112111555555555111		11555551
1551	1112	1 121 1 11 1 15555555111	0	15555551
1551	12 2	132 2 1155555111	0	115555551
1551	222 0	322 115555111	121	155555551
1551	12 0	1 1555551	131	1255555551
1551	2 0	115555511	1	155555551
1551	2 0	1155555551	0	1 155555551
1551	2	11555555551		21155555551
1551	1 0	115555555551		15555555551
1551		11511115555521	1	115555555551
1551	1 0	11111 1155511	2	155555555551
1551	131	111 15111	2	155555555551

1551	121			1121	1	111	1	2	1155555555112
1551	11			111	1	221	11	1	2 15555555551 2
1551	12	0	1	21	121	11	1111	2	1555555555112
1551	1		12	22	1511111111551			2	11555555555551
1551	1			2	1555551115511	1			15555555555551
1551	2	0	0	22	12555551	15551		1	15555555555551
1551	1			1	1555511	11511		2	115555555555551
1551		0	0	21	155551	1	151	2	155555555555551
1551				2	15555112	151		2	155555555555551
1551		1	1	1	1255555511111			2	155555555555551
1551		2	22		11511111212				2115555555555551
1551	0		1	12	111	2	1		15555555111555551
1551		0	0	0	111	121			155555551 1555551
1551			0		11111111				155555551 1555551
1551		0			115551				155555551 1555511
1551					15551				211111111 155511
11521		1	12		0	1155511		2	11 115511
	151	0		1	1				155555111 2111 15511
12	1511		1			15555555111	155111		1511
22	1511		1			15555555551	155551		1151
2	151		0			1155555555511	155511		1511
2	1521	0		0		115555555555511	15551		12151
2	151			121		155555555555551	155511		1551
2	1511				0	155555555555551	115551		1511
21	1511			11		155555555555551	111111151		
11	151		0			11555555555555511	111511		
11	151					15555555555555551	151		
11	151					115555555555555551	211		
11	151					1155555555555555511	1		
11	151				0	155555555555555551			
11	111		0			12111111111111111111			

* 整理過後得到的結果圖 *

