Computer-vision-Homework 6

Yokoi Connectivity Number

Due date：23 Nov 2021

Programming language: python 3.9.7

Import lib:

- Opencv: to read and write the image file
- Numpy: to work with the arrays

Original image: lena.bmp

[512(width),512(height),1channel(cv2.IMREAD_GRAYSCALE)]

Code explanation:

```
lena = cv2.imread('lena.bmp', cv2.IMREAD_GRAYSCALE)
```

- imread to load a file with cv2.IMREAD_GRAYSCALE(or using 0 as the parameter of the function)

Q1：matrix of Yokoi connectivity number

```
def DownSample(image, DownSampleSize):
    DownSampleImage = np.zeros((DownSampleSize, DownSampleSize))
    # return DownSampleImage
    DownSamplelength = int(len(image) / DownSampleSize)
    for i in range(len(DownSampleImage)):
        for j in range(len(DownSampleImage[i])):
            DownSampleImage[i][j] = image[DownSamplelength *
i][DownSamplelength * j]
    return DownSampleImage


def Binarize(image):
    BinarizeImage = np.zeros(image.shape)
    for i in range(len(image)):
        for j in range(len(image)):
            BinarizeImage[i][j] = 0 if image[i][j] < 128 else 255
    return BinarizeImage
```

First I Binarized and Downsampling the lena.bmp from 512 to 64 with threshold value 128.

```
def Yokoi(image):
    YokoiImage = np.zeros(image.shape)
    for i in range(len(image)):
        for j in range(len(image[i])):
            if image[i][j] == 0:
                YokoiImage[i][j] = 0
                continue
            r, q = 0, 0
            # a1
            if j + 1 < len(image[i]) and image[i][j] == image[i][j +
1]:
                q += 1
            if i - 1 >= 0 and j + 1 < len(image[i]) and image[i][j] ==
image[i][j + 1] == image[i - 1][j] == image[i - 1][j + 1]:
                q -= 1
                r += 1
            # a2
            if i - 1 >= 0 and image[i][j] == image[i - 1][j]:
```

```
                q += 1
            if j - 1 >= 0 and i - 1 >= 0 and image[i][j] == image[i -
1][j] == image[i - 1][j - 1] == image[i][j - 1]:
                q -= 1
                r += 1
            # a3
            if j - 1 >= 0 and image[i][j] == image[i][j - 1]:
                q += 1
            if i + 1 < len(image) and j - 1 >= 0 and image[i][j] ==
image[i][j - 1] == image[i + 1][j] == image[i + 1][
                j - 1]:
                q -= 1
                r += 1
            # a4
            if i + 1 < len(image) and image[i][j] == image[i + 1][j]:
                q += 1
            if j + 1 < len(image[i]) and i + 1 < len(image) and
image[i][j] == image[i + 1][j] == image[i][j + 1] == image[i + 1][j +
1]:
                q -= 1
                r += 1
            if r == 4:
                YokoiImage[i][j] = 5
            else:
                YokoiImage[i][j] = q
    return YokoiImage
```

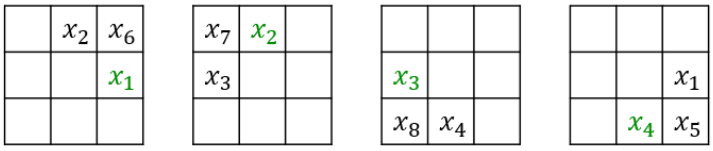
According to the definition of Yokoi connectivity number

- for 4-connectivity

$$h(b,c,d,e) = \begin{cases} q & if\ b = c\ and\ (d \neq b \vee e \neq b) \\ r & if\ b = c\ and\ (d = b \wedge e = b) \\ s & if\ b \neq c \end{cases}$$

- $a_1 = h(x_0, x_1, x_6, x_2)$
  $a_2 = h(x_0, x_2, x_7, x_3)$
  $a_3 = h(x_0, x_3, x_8, x_4)$
  $a_4 = h(x_0, x_4, x_5, x_1)$

Corner Neighborhood
(for corresponding x_i)

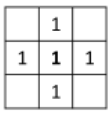


6.2.5.1 Yokoi Connectivity Number

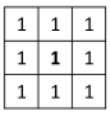
- for 4-connectivity

$$h(b,c,d,e) = \begin{cases} q & if\ b = c\ and\ (d \neq b\ \vee e \neq b) \\ r & if\ b = c\ and\ (d = b \wedge e = b) \\ s & if\ b \neq c \end{cases}$$

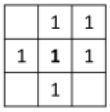
$$f(a_1,a_2,a_3,a_4) = \begin{cases} 5 & if\ a_1 = a_2 = a_3 = a_4 = r \\ n & where\ n = number of\{a_k | a_k = q\}, otherwise \end{cases}$$



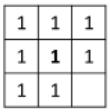
$a_1 = q$
$a_2 = q$
$a_3 = q$
$a_4 = q$
$f(a_1, a_2, a_3, a_4) = 4$
(crossing)



$a_1 = r$
$a_2 = r$
$a_3 = r$
$a_4 = r$
$f(a_1, a_2, a_3, a_4) = 5$
(interior)



$a_1 = r$
$a_2 = q$
$a_3 = q$
$a_4 = q$
$f(a_1, a_2, a_3, a_4) = 3$
(branching)



$a_1 = r$
$a_2 = r$
$a_3 = r$
$a_4 = q$
$f(a_1, a_2, a_3, a_4) = 1$
(edge)

According to the formula above, we can calculate the matrix of Yokoi Connectivity Number as above and get the answer.