Computer Vision HW6, Yokoi Connectivity Report

tags: NTU CS Computer Vision Writeup Report

NTU CSIE, R08922024, Alfons Hwu
Prequisites and env as the following

```
pure vim + python3 only
```

a, Yokoi-connectivity

Just brute-forcely implement all the possibilities for the answer. (Answer as attatched `yokoi_ans.txt`)

```
####### yokoi h op ####

def h(b, c, d, e):
    if b == c and (d != b or e != b):
        return 'q'
    if b == c and (d == b and e == b):
        return 'r'

    return 's'

####### yokoi core ####
def yokoi_core(img_in):
    row, col = img_in.shape

    # down sampling image
    img_down = np.zeros((64, 64), np.int)
    row, col = img_down.shape
    for i in range(row):
        for j in range(col):
            img_down[i, j] = img_in[8 * i, 8 * j]

    for i in range(row):
        for j in range(col):
            if img_down[i, j] > 0:
                if i == 0:
                    if j == 0:
                    # top-left
                        x7, x2, x6 = 0, 0, 0
                        x3, x0, x1 = 0, img_down[i, j], img_down[i, j + 1]
                        x8, x4, x5 = 0, img_down[i + 1, j], img_down[i + 1, j
                    elif j == col - 1:
                    # top-right
                        x7, x2, x6 = 0, 0, 0
                        x3, x0, x1 = img_down[i, j - 1], img_down[i, j], 0
                        x8, x4, x5 = img_down[i + 1, j - 1], img_down[i + 1, j
                    else:
                    # top-row
                        x7, x2, x6 = 0, 0, 0
                        x3, x0, x1 = img_down[i, j - 1], img_down[i, j], img_d
                        x8, x4, x5 = img_down[i + 1, j - 1], img_down[i + 1, j
                elif i == img_down.shape[0] - 1:
                    if j == 0:
                    # bottom-left
                        x7, x2, x6 = 0, img_down[i - 1, j], img_down[i - 1, j
                        x3, x0, x1 = 0, img_down[i, j], img_down[i, j + 1]
                        x8, x4, x5 = 0, 0, 0
                    elif j == col - 1:
                    # bottom-right
                        x7, x2, x6 = img_down[i - 1, j - 1], img_down[i - 1, j
                        x3, x0, x1 = img_down[i, j - 1], img_down[i, j], 0
```

```
                x8, x4, x5 = 0, 0, 0
            else:
            # bottom-row
                x7, x2, x6 = img_down[i - 1, j - 1], img_down[i - 1, j
                x3, x0, x1 = img_down[i, j - 1], img_down[i, j], img_d
                x8, x4, x5 = 0, 0, 0
        else:
            if j == 0:
            # leftmost-row
                x7, x2, x6 = 0, img_down[i - 1, j], img_down[i - 1, j
                x3, x0, x1 = 0, img_down[i, j], img_down[i, j + 1]
                x8, x4, x5 = 0, img_down[i + 1, j], img_down[i + 1, j
            # rightmost-column
            elif j == col - 1:
                x7, x2, x6 = img_down[i - 1, j - 1], img_down[i - 1, j
                x3, x0, x1 = img_down[i, j - 1], img_down[i, j], 0
                x8, x4, x5 = img_down[i + 1, j - 1], img_down[i + 1, j
            #the rest, inner
            else:
                x7, x2, x6 = img_down[i - 1, j - 1], img_down[i - 1, j
                x3, x0, x1 = img_down[i, j - 1], img_down[i, j], img_d
                x8, x4, x5 = img_down[i + 1, j - 1], img_down[i + 1, j

        a1 = h(x0, x1, x6, x2)
        a2 = h(x0, x2, x7, x3)
        a3 = h(x0, x3, x8, x4)
        a4 = h(x0, x4, x5, x1)

        if a1 == 'r' and a2 == 'r' and a3 == 'r' and a4 == 'r':
            ans = 5
            print(ans, end='')
        else:
            ans = 0
            for a_i in [a1, a2, a3, a4]:
                if a_i == 'q':
                    ans += 1

            if ans != 0:
                print(ans, end='')
            else:
                print(' ', end='')

    else:
        # background
        print(' ', end='')

    if j == col - 1:
        # new line
        print('')
```