# R10922123 周昱豪

#### Code

## Part(a):

把每個 pixel 抓出來後只檢查值是 255 的, 然後把 kernel 範圍內的都改成 255。

#### Part(b):

有分成兩個部分,用 mode 來區分,mode = 0 時是給正常的 erosion 使用,會去 判斷現在這個 pixel 是不是 255; mode = 1 時是給 hit-and-miss 的 complement image 使用的,因為 kernel k 他的判別方式不會分黑跟白,所以就不需要判斷當前 pixel value 是否為 255。

```
def erosion(a, b, mode):
    row_a, col_a = a.shape
    img = np.zeros(a.shape, dtype='int32')
    for a_i in range(row_a):
         for a_j in range(col_a):
             if a[a_i, a_j] == 0xff and not mode:
                  img[a_i, a_j] = 0xff
                  flag = 1
                  for lis in b:
                      b_i, b_j = lis
                       if (a_i + b_i) >= row_a \text{ or } (a_i + b_i) < 0 \text{ or } \setminus
                           (a_j + b_j) >= col_a or (a_j + b_j) < 0 or \
                           a[a_i+b_i, a_j+b_j] != 0xff:
                           flag = 0
                  if flag == 0:
                       img[a_i, a_j] = 0
             elif mode == 1:
                  flag = True
                  for lis in b:
                      b_i, b_j = lis
                       if (a_i + b_i) >= row_a \text{ or } (a_i + b_i) < 0 \text{ or } \setminus
                           (a_j + b_j) >= col_a \text{ or } (a_j + b_j) < 0 \text{ or } 
                           a[a_i+b_i, a_j+b_j] != 0xff:
flag = False
                           break
                  if flag:
                      img[a_i, a_j] = 0xff
    return img
```

### Part(c)

```
def opening(a, b):
    cv2.imwrite("opening_lena.bmp", dilation(erosion(a, b, 0), b))
```

### Part(d)

```
def closing(a, b):
    cv2.imwrite("closing_lena.bmp", erosion(dilation(a, b), b, 0))
```

### Part(e):

要注意的是這邊的 complement 在跟 kernel k 做 erosion 時,不可以只做白點,黑點也要考慮進去,所以才在 erosion 那加上 mode 來判斷現在是要做哪項 task,不然會得到一張全黑的圖。

```
def hit_and_miss(img, j, k):
    img_comp = -img + 0xff
    return (((erosion(img, j, 0) + erosion(img_comp, k, 1)) / 2) == 0xff) * 0xff
```

#### Result

Dilation



Erosion



Opening



Closing



Hit-andmiss transform

