Computer Vision I

Homework 5 - Mathematical Morphology - Gray Scaled　B04902090 資工四 施長元

Usage of the full code:

`python main.py [Image_path]`

After the code exit, output files will be in the directory where you execute the code.

Environment: Python3.6 on Windows Linux Subsystem (Ubuntu 16.04)

Contents:

Kernel: Octagonal 3-5-5-5-3 kernel

Write programs which do gray-scale morphological:

- Dilation

  兩層迴圈找到 kernel 涵蓋到的所有 pixel 中之最大值，再將同一批 pixel 設定為此最大值

```
def dilation(img_o):
  img_t = np.zeros(img_o.shape, dtype=np.int32)
  for i in range(img_o.shape[0]):
    for j in range(img_o.shape[1]):
      if img_o[i, j] > 0:
        max_l = 0
        # Find Max
        for x, y in zip(kernel0_x, kernel0_y):
          if i+x-2 > -1 and i+x-2 < img_o.shape[0] and j+y-2 > -1 \
            and j+y-2 < img_o.shape[1]:
            max_l = max(max_l, img_o[i+x-2, j+y-2])
        # Propagate
        for x, y in zip(kernel0_x, kernel0_y):
          if i+x-2 > -1 and i+x-2 < img_o.shape[0] and j+y-2 > -1 \
            and j+y-2 < img_o.shape[1]:
            img_t[i+x-2, j+y-2] = max_l
  return img_t
```

Results: 左圖

▲Dilation

▲Erosion

- Erosion

  兩層迴圈找到 kernel 涵蓋到的所有 pixel 中，若任何 pixel 為 0 或是超出 boundaries，則跳過這個 kernel(原先全部都是 0)；否則，找到最小值並套用在同一批 pixels 上

```
def erosion(img_o):
  img_t = np.zeros(img_o.shape, dtype=np.int32)
  for i in range(img_o.shape[0]):
    for j in range(img_o.shape[1]):
      min_l = 256
      for x, y in zip(kernel0_x, kernel0_y):
        # Boundaries, confirm all > 0, and find the Min
        if i+x-2 < 0 or i+x-2 > img_o.shape[0]-1 or j+y-2 < 0 \
          or j+y-2 > img_o.shape[1]-1 or img_o[i+x-2, j+y-2] == 0:
          min_l = -1; break
        else: min_l = min(min_l, img_o[i+x-2, j+y-2])
      # Propagate
      if min_l != -1:
        for x, y in zip(kernel0_x, kernel0_y):
          img_t[i+x-2, j+y-2] = min_l
  return img_t
```

  Result: 上頁圖右側

- Opening

  先 erosion 再 dilation $(B \circ K = (B \ominus K) \oplus K)$

```
cv2.imwrite("hw5_opening.bmp", dilation(erosion(img)))
```

  Result: 下圖左側

- Closing

  先 dilation 再 erosion $(B \cdot K = (B \oplus K) \ominus K)$

```
cv2.imwrite("hw5_closing.bmp", erosion(dilation(img)))
```

  Result: 下圖右側

▲Opening

▲Closing