

# 數位影像處理

## Digital Image Processing

### 作業三 HW3b

電機所 R11921038 江讀晉

2022/11/23

### Execution

- Problem 3 : `python p3.py` , 實作成果參考 `3-0_overlap.jpg` 、 `3-1_isolate.jpg` 以及 `3-2_broader_boundary.jpg`

### Synopsis

- 利用 erosion 的原理，小於 structure element (SE) 的 brick 會直接被侵蝕到不見。
- 選取適當大小的 SE 做 opening，則可將小於 SE 的部分剔除，並保留大於 SE 的部分。

### Problem 3. Morphology

- Assume the particles have a rectangular shape and are of the same size.

首先，我想先找出 brick size，以方便後續題目使用。利用 erosion 的原理，小於 structure element 的 brick 會直接被侵蝕到不見。因此，我將 brick size 限縮在一個適當的上下界，用 grid search 的方法評估，而評估方式則為當 erosion 至前景完全消失，則找出 brick size。整體流程概述如 Fig.1 所示。

計算出的 brick size 大約為 23 x 28。另外，我發現似乎並非每一塊 brick 都一樣大，存在一些誤差（也可能是 OpenCV 的 erosion 方法和我的想法並不完全一致），因此我僅將 brick size 作為參考來調整以下兩題的參數。

	<b>get_brick_size:</b> input - image / output - size of the brick
1	size 1 to N ← meshed grid of brick size
2	<b>for_loop</b> from size1 to sizeN <b>do</b>
3	kernel ← build 1s array with the given size
4	do erosion
5	if there is no foreground
6	return proper size
7	<b>end</b>

Fig.1 找出 brick size 概述

- a. Design a morphological algorithm that takes the image bricks.tif as input and generates an output image consisting of only overlapping particles.

承接上述，利用 erosion 的原理，小於 structure element (SE) 的 brick 會直接被侵蝕到不見，而比 SE 大的部分則會有剩餘，此時可以再用相同的 SE 做 dilation 就可以還原出原本的 brick。也就是說，選取適當大小的 SE 做 opening，則可將小於 SE 的部分剔除，保留大於 SE 的部分。

因此，將原影像 zero-padding 後，我使用橫條和長條形狀的 SE 分別做 erosion，並將原圖和結果分別相減，得到 overlapping 被破壞的影像（如 Fig.2、Fig.3 所示）。之後，使用比 brick size 小一些的 kernel 對前述結果實作 opening 得到可能為 non-overlapping 的部分，將其去除（如 Fig.4、Fig.5 所示）並結合所有結果，即得到最終 overlapping 的區域（如 Fig.6 所示）。

順帶一提，因為有些 brick 貼著邊緣，如果沒有加上 padding 的話，erosion 時無法將其剔除。因此，在一開始加上 zero-padding，之後再擷取原圖尺寸即可。

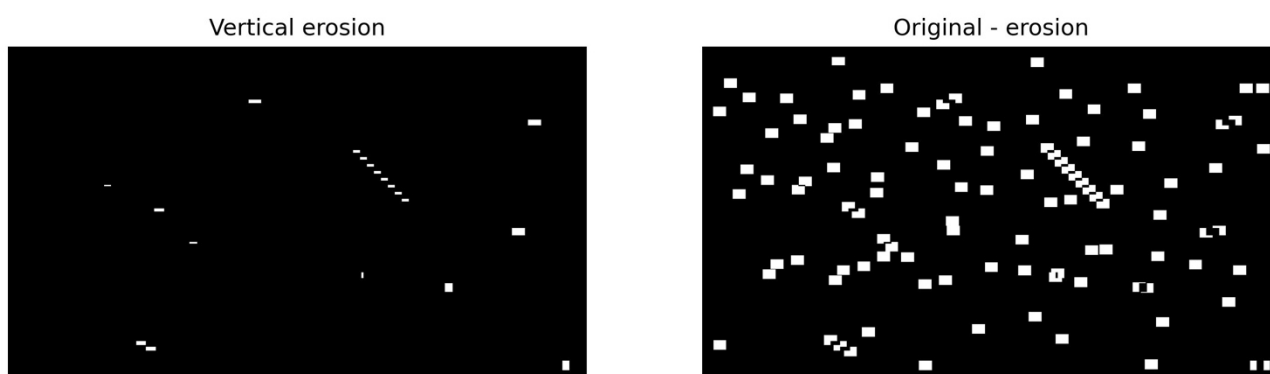


Fig.2 左圖為使用大小 1 x 31 的 kernel 實作 erosion，右圖將原圖減去左圖，可看出部分 brick 交集的部分已剔除



Fig.3 左圖為使用大小 26 x 1 的 kernel 實作 erosion，右圖將原圖減去左圖，可看出部分 brick 交集的部分已剔除

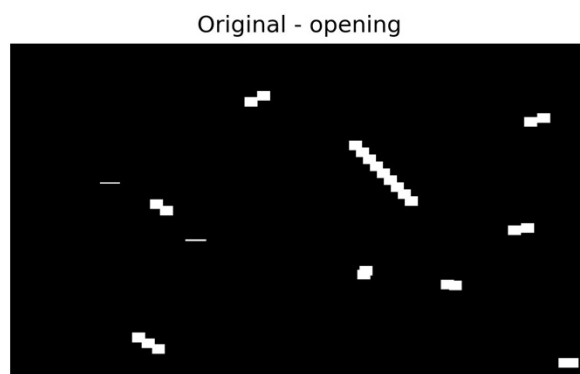
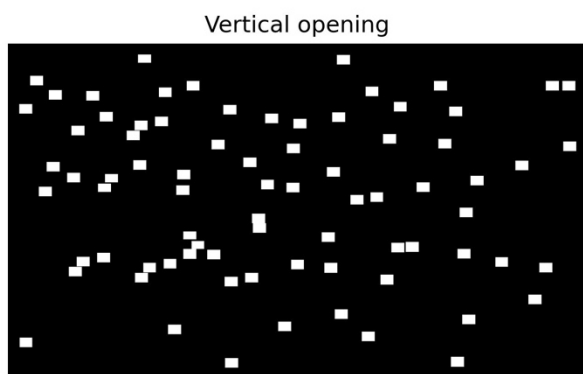


Fig.4 左圖為使用大小  $18 \times 24$  的 kernel 對 Fig.2 實作 opening，右圖將原圖減去左圖，得到 overlapping 的區域

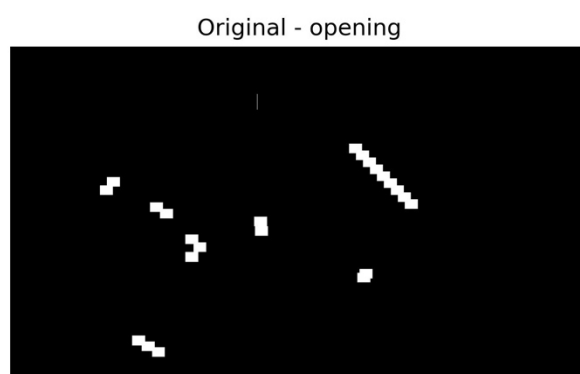
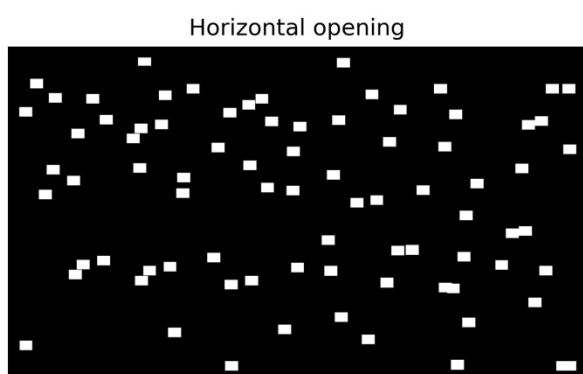


Fig.5 左圖為使用大小  $18 \times 24$  的 kernel 對 Fig.3 實作 opening，右圖將原圖減去左圖，得到 overlapping 的區域

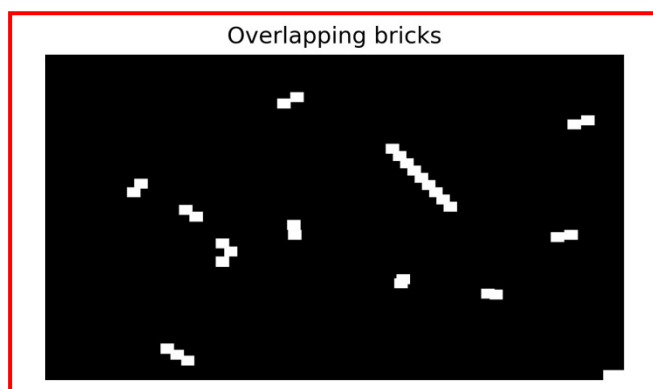
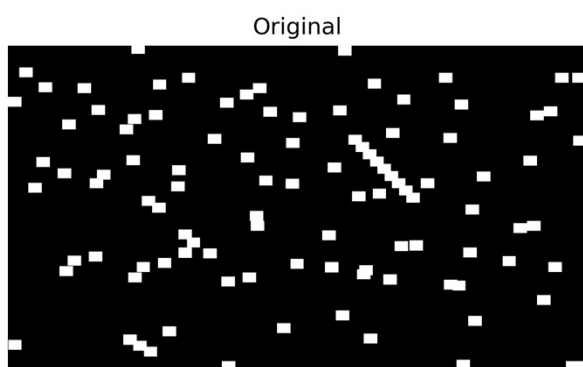


Fig.6 左圖為原圖，右圖將 Fig.4 右圖和 Fig.5 右圖的結果合併並移除 padding 後的成果

- b. Design a morphological algorithm to extract the isolated particles that do not touch the border of the image bricks.tif.

承接 section a 所述，將沒有 padding 的原圖實作 Fig.2 至 Fig.5 的流程無法將接觸邊界的 brick 移除。利用此特性，我使用沒有 padding 整體的圖片找出 overlapping 的區域，再減去 Fig.6 右圖得到的最終 overlapping 區域，即得到 isolated bricks。如 Fig.7 所示。

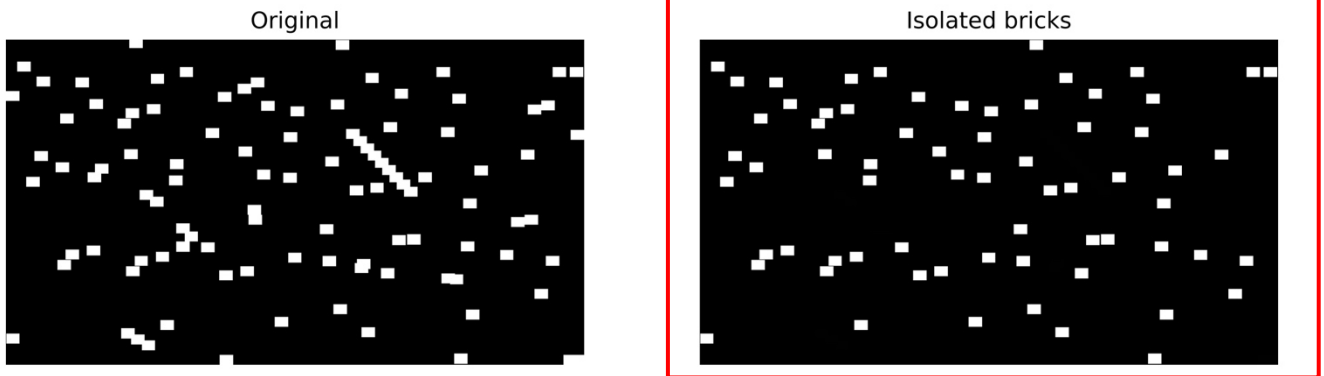


Fig.7 左圖為原圖，右圖為沒有碰觸邊界的 isolated bricks

然而，根據 Fig.7 右圖，可發現有四個 bricks 和邊界之間的空隙極小（小於 1pixel），似乎和沒碰觸邊界的感覺有所矛盾。因此，我將圖片邊緣寬度為 10 的區域填 0，再對其做 opening，得到 Fig.8 的結果。

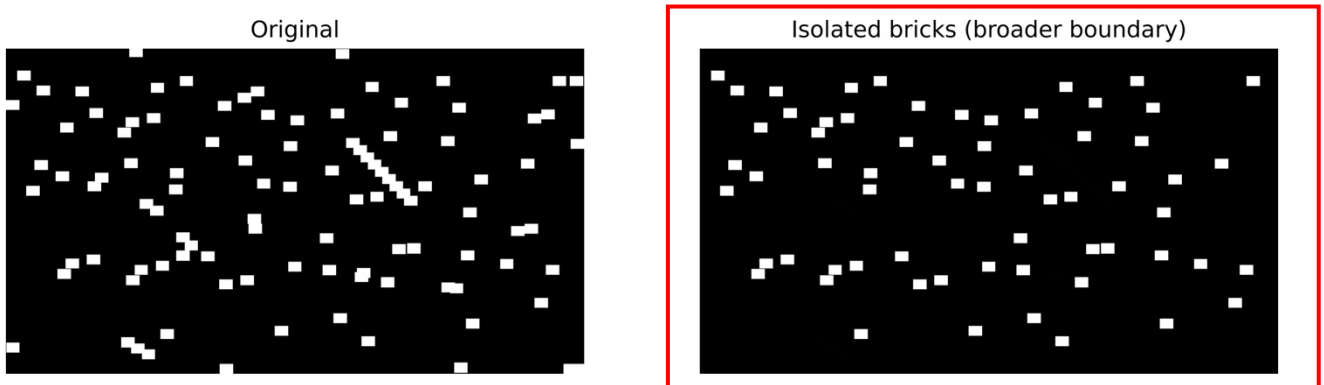


Fig.8 左圖為原圖，右圖為沒有碰觸邊界的 isolated bricks，且黑色 margin 區域明顯

## Reference

- [1] OpenCV. Morphological Transformations. 2022.  
[https://docs.opencv.org/4.x/d9/d61/tutorial\\_py\\_morphological\\_ops.html](https://docs.opencv.org/4.x/d9/d61/tutorial_py_morphological_ops.html)
- [2] Jason Chen。【影像處理】形態學 Morphology。2019 年。  
<https://jason-chen-1992.weebly.com/home/-morphology>