

# 高等數位影像處理

## 作業#6

姓名：\_\_\_\_\_巫伯銘\_\_\_\_\_

學號：\_\_\_\_\_111318096\_\_\_\_\_

指導老師：\_\_\_\_\_張陽郎 教授\_\_\_\_\_

## 1.

### Figure

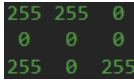
01\_words\_morphology.png

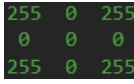
**Morphology**  
Morphology  
**Morphology**  
Morphology  
**Morphology**  
Morphology

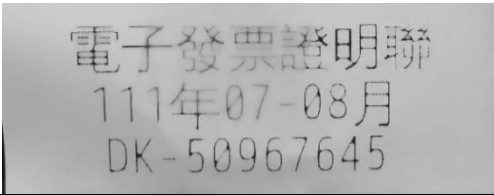
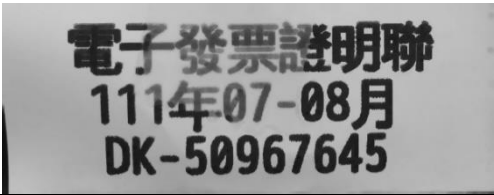
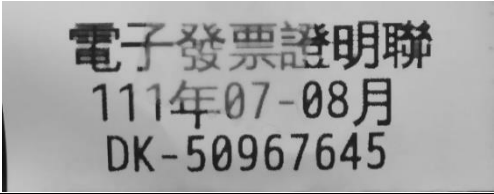
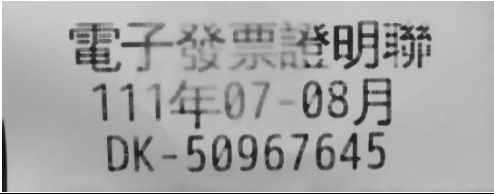
### Discussion

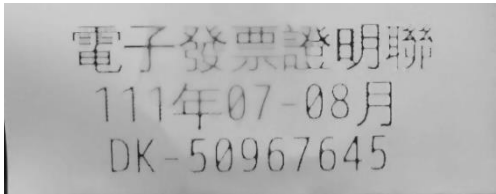
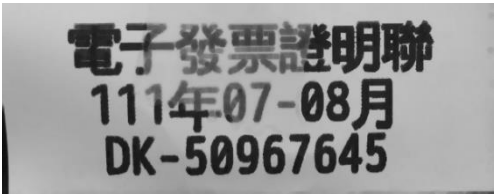
先使用  的 kernel 對原圖做侵蝕→膨脹→侵蝕；

再使用  的 kernel 做一次膨脹；

再使用  的 kernel 做一次侵蝕把大部分之雜點清除；

最後再使用  的 kernel 做一次膨脹把一些縫隙補起來。

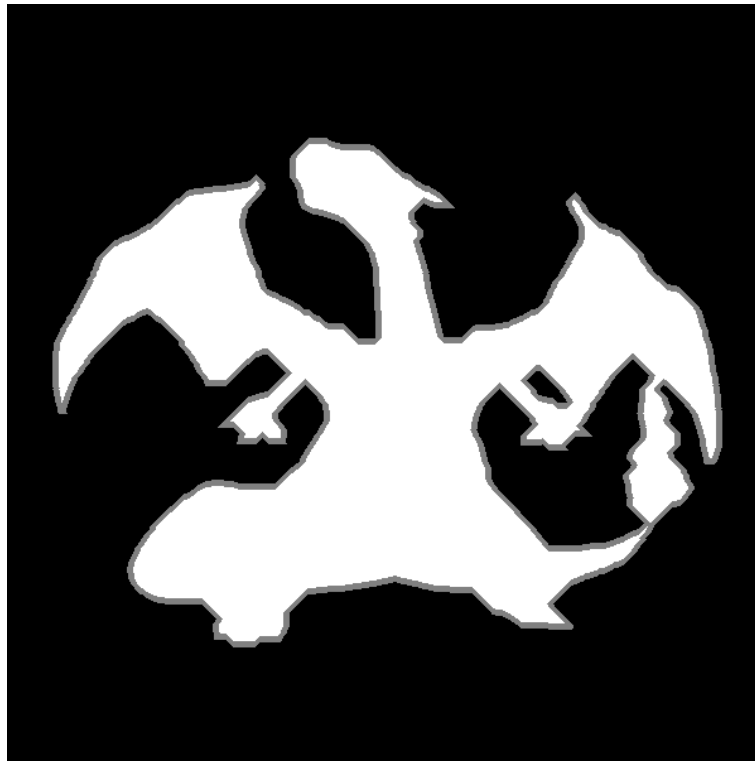
2(1)	
Figure	
02_check_erode_SE_5x5.png 	03_check_dilate_SE_5x5.png 
04_check_open_SE_5x5.png 	05_check_close_SE_5x5.png 
Discussion	
<p>比較以上四種做法的結果，侵蝕後字體變細；膨脹後字體變粗；open 則是相較於 close 字體更加完整了。</p>	

<b>2(2)</b>	
Figure	
06_check_erode_SE_5x5_90.png 	07_check_dilate_SE_5x5_90.png 
Discussion	
<p>旋轉 90 度後的 SE_5x5 與原先未旋轉的侵蝕與膨脹結果基本上一模一樣，  肉眼上是觀察不出來的。</p>	

### 3.

#### Figure

08\_monster\_trimap.png



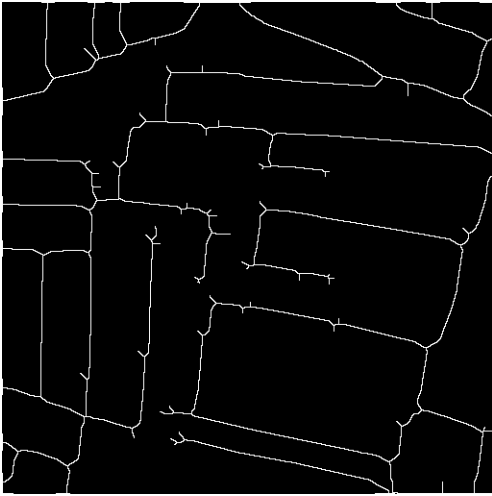
#### Discussion

本題先將 monster\_600.raw 做二值化，門檻設為 245，接下來使用十字的 kernel 做膨脹直到內部變成實心白色為止，最後再將膨脹前後有變化的位置寫成灰色(128)，結果如圖所示。

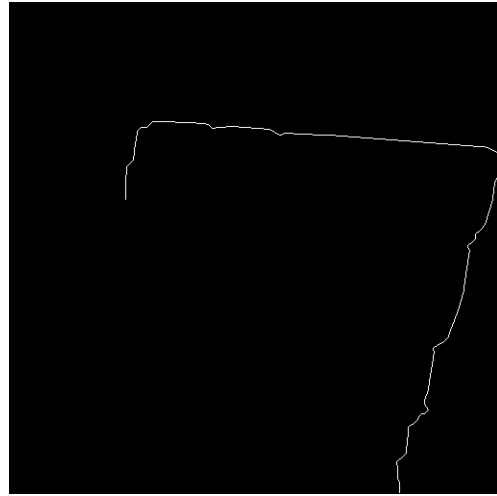
#### 4.

Figure

09\_map\_thin.png



10\_map\_thin\_path.png



(step = 814)

Discussion

```
int kernel0[3][3] = {
    {0, 0, 0},
    {-1, 255, -1},
    {255, 255, 255}
};
int kernel1[3][3] = {
    {-1, 0, 0},
    {255, 255, 0},
    {255, 255, -1}
};
int kernel2[3][3] = {
    {255, -1, 0},
    {255, 255, 0},
    {255, -1, 0}
};
int kernel3[3][3] = {
    {255, 255, -1},
    {255, 255, 0},
    {-1, 0, 0}
};
```

```
int kernel4[3][3] = {
    {255, 255, 255},
    {-1, 255, -1},
    {0, 0, 0}
};
int kernel5[3][3] = {
    {-1, 255, 255},
    {0, 255, 255},
    {0, 0, -1}
};
int kernel6[3][3] = {
    {0, -1, 255},
    {0, 255, 255},
    {0, -1, 255}
};
int kernel7[3][3] = {
    {0, 0, -1},
    {0, 255, 255},
    {-1, 255, 255}
};
```

以上 8 個 kernel0 ~ kernel7 為本題使用的 kernels，倘若 kernel 內 0 與 255 之值都完全與原圖符合，則代表該點還太寬，因此將其寫 0。如此按照 0 到 7 的順序對原圖做一次將得到邊緣的一次細化，本題做了 20 次，最後只剩下一格 pixel 如上圖所示。

本題使用 D8 尋找最短路徑，步數為 814，結果如上圖所示。