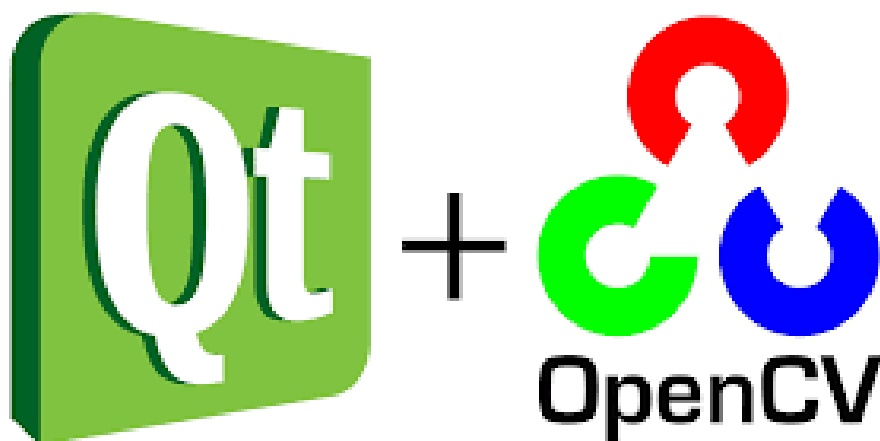


# 影像處理程式開發平台作業



使用 Qt + OpenCV 實作

資工三甲

4A8G0039

楊孟繁學

[https://github.com/4A8G0039/HW\\_OpenCV](https://github.com/4A8G0039/HW_OpenCV)

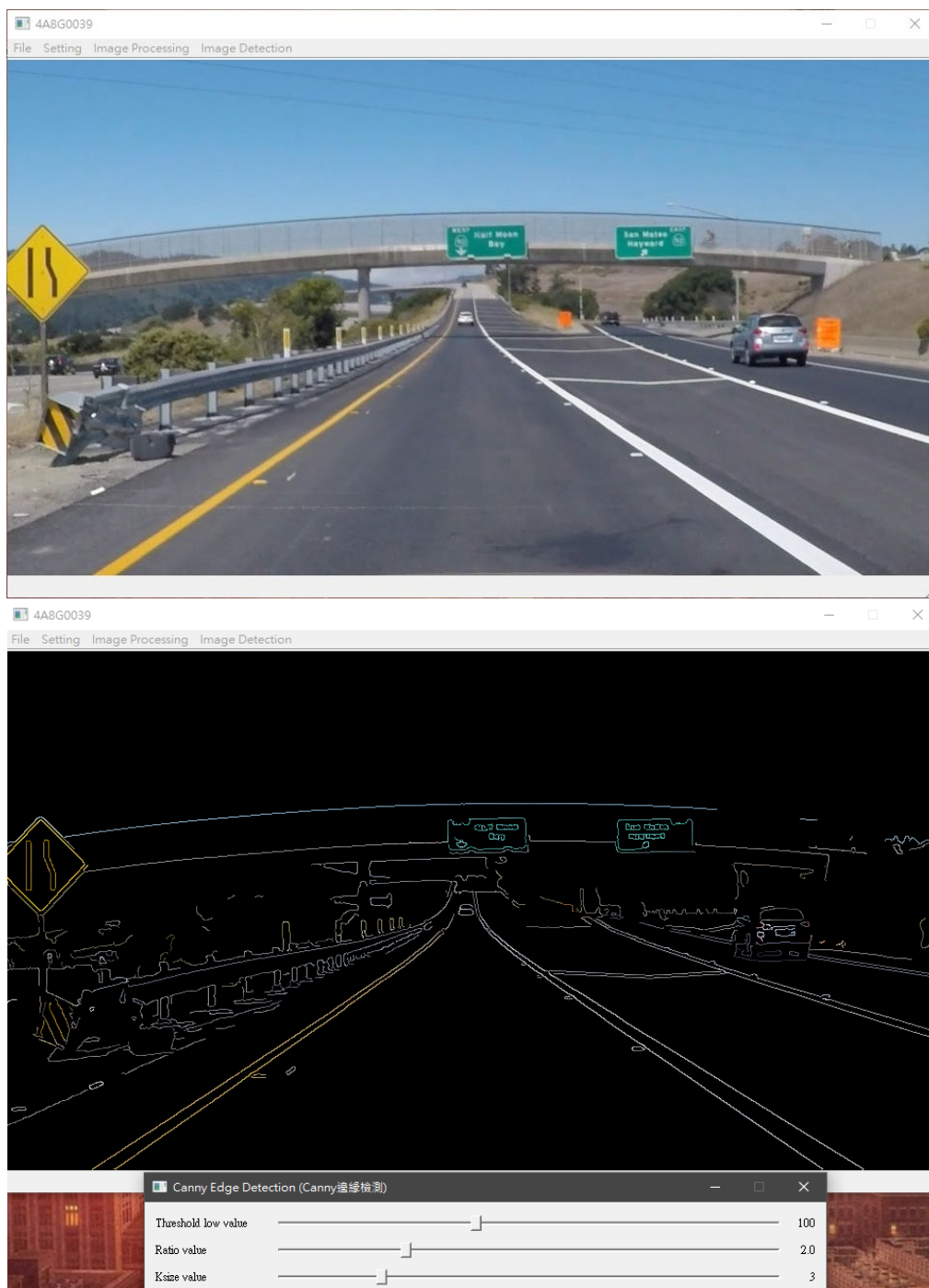
## Canny Edge Detection (Canny 邊緣檢測) :

使用 `cv2.Canny()` 抓取物件邊緣。

Threshold low value : 最低閾值

Ratio value : 高低閾值比

Ksize value : 模糊內核大小



## Hough Line Transform (霍夫轉換) :

使用 `cv2.Canny()` 抓取物件邊緣後再使用 `cv2.HoughLines()` 或 `cv2.HoughLinesP()` 找出圖中的直線。

### Canny Edge Detection :

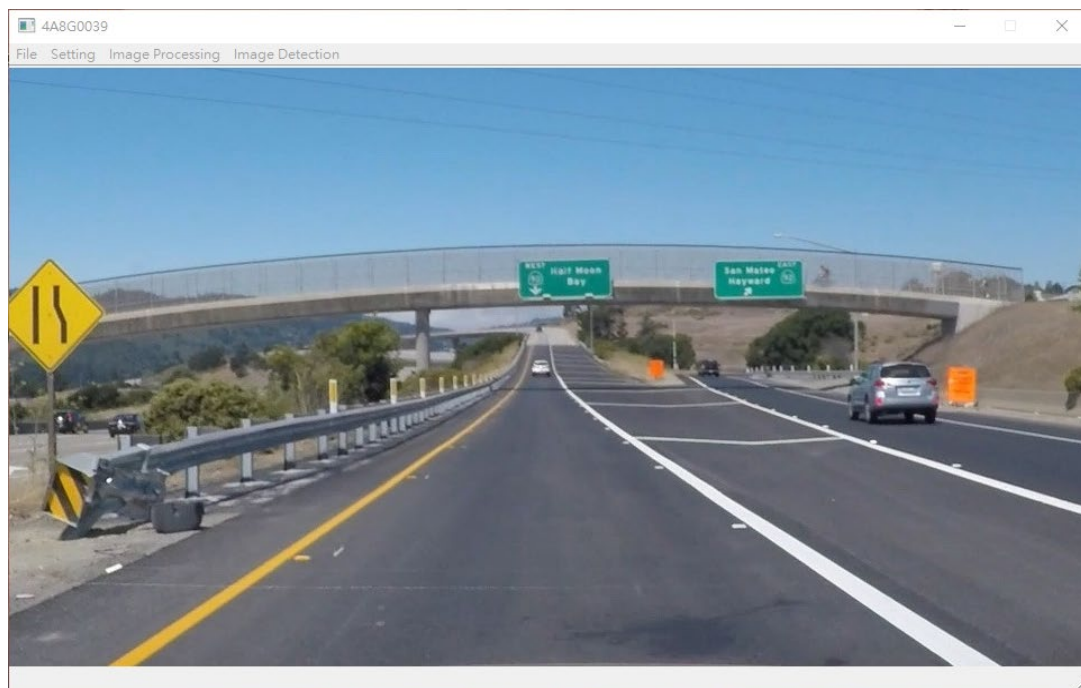
Threshold low value : 最低閾值

Ratio value : 高低閾值比

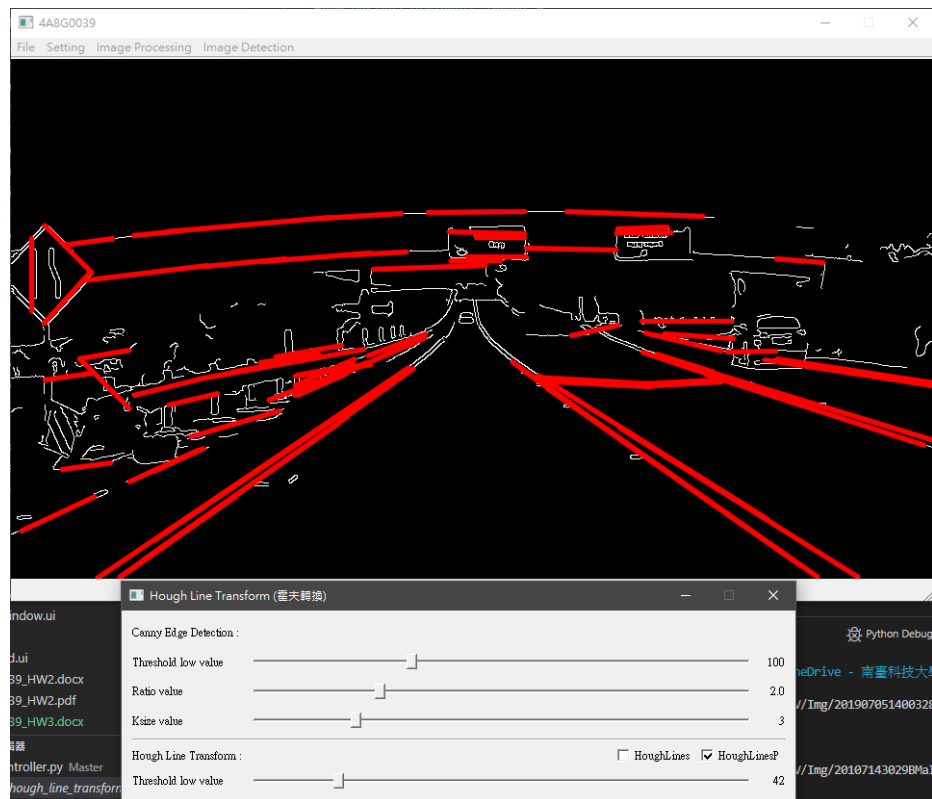
Ksize value : 模糊內核大小

### Hough Line Transform :

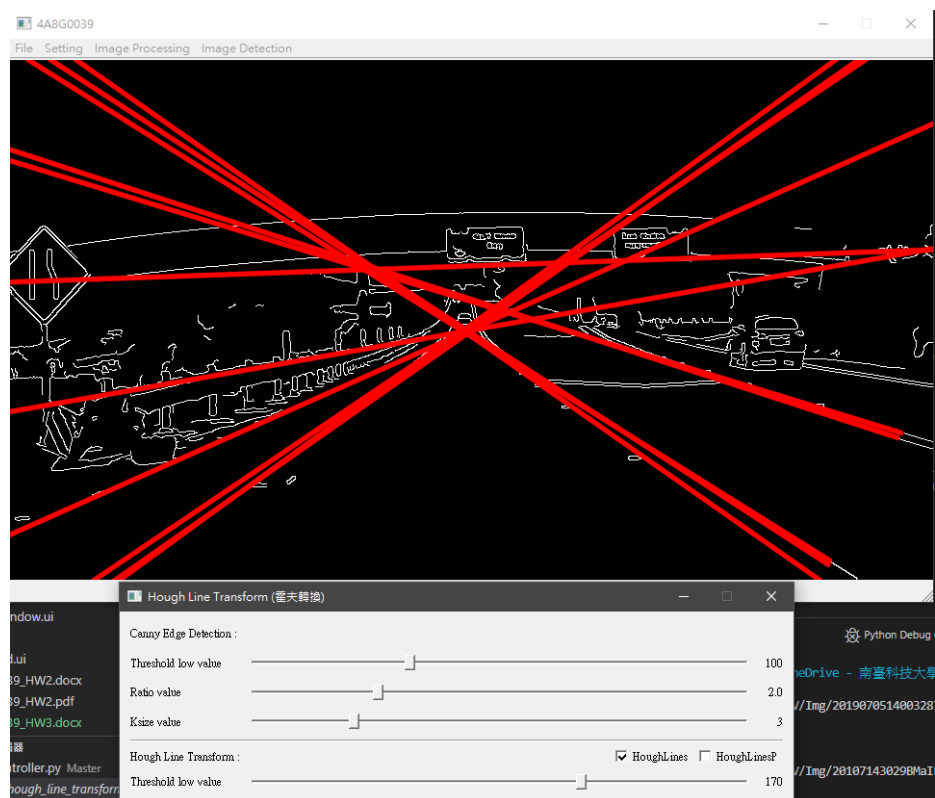
Threshold low value : 最低閾值



## HoughLinesP :



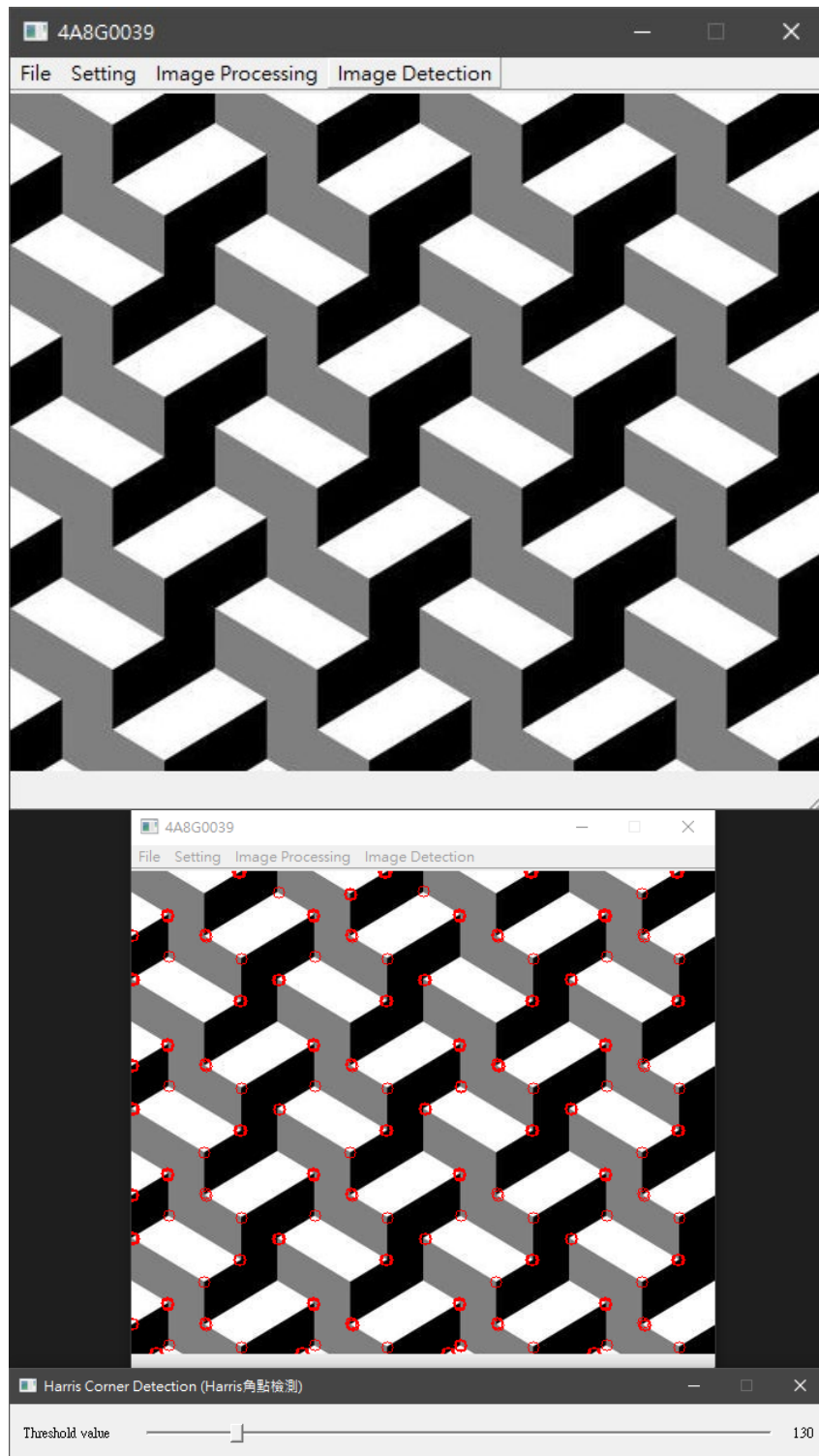
## HoughLines :



## Harris Corner Detection (Harris 角點檢測) :

使用 `cv2.cornerHarris()` 找出圖中的角並圈出來。

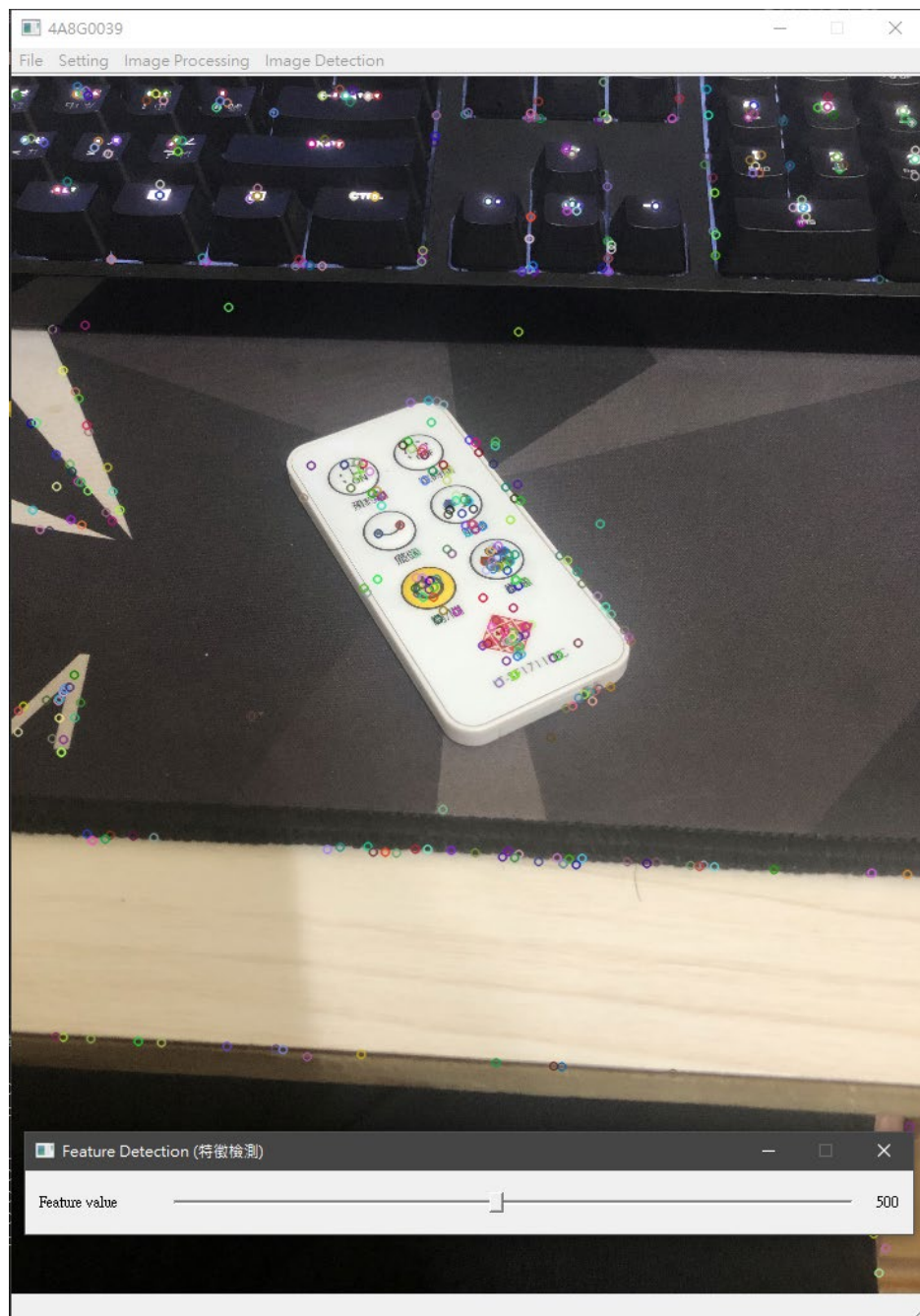
Threshold value : 閾值



## Feature Detection (特徵檢測) :

使用 `cv2.SIFT_create()` 找出圖中的特徵點並圈出來。

Feature value : 特徵點數量

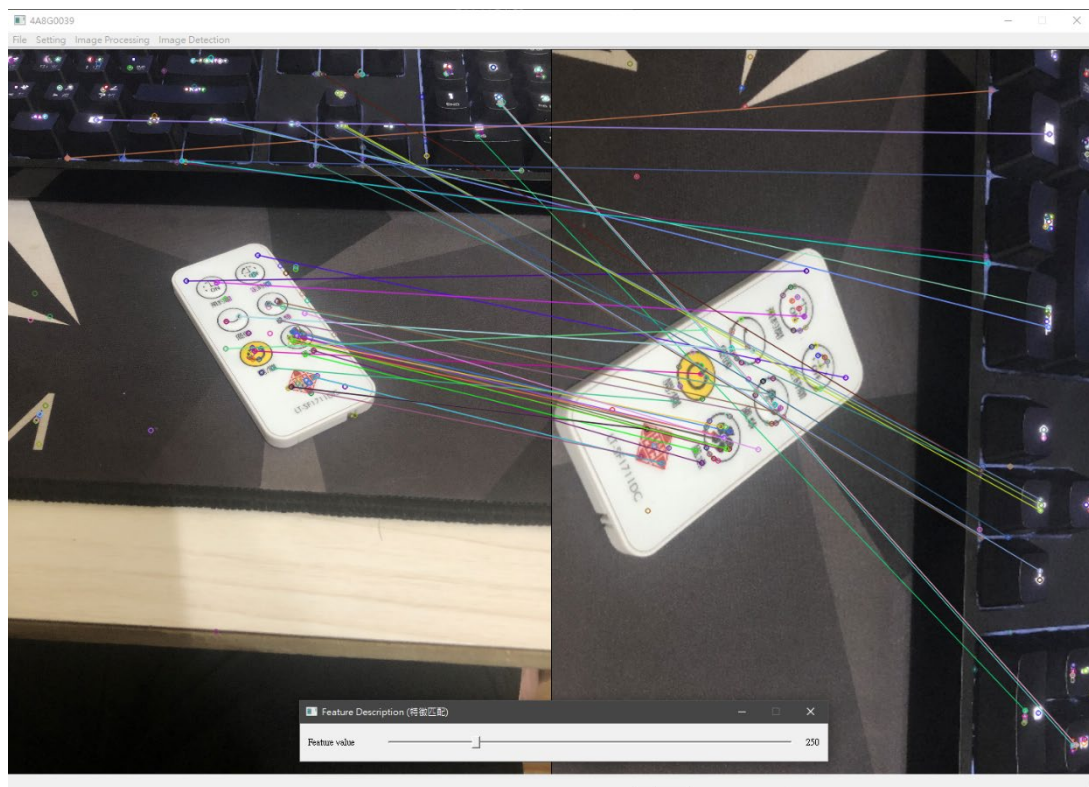




## Feature Description (特徵匹配) :

使用 `cv2.SIFT_create()` 找出圖中的特徵點並圈出來，  
再用 `cv2.BFMatcher()` 比對兩張圖的特徵點，  
最後再把權重較高的對應點連起來。

Feature value : 特徵點數量



## Finding contours (尋找輪廓) :

使用 `cv2.Canny()` 抓取物件邊緣後再使用 `cv2.findContours()` 找出物件的輪廓。

找出輪廓後就可：

- 使用 `cv2.convexHull()` 找出物件的凸包(最外框)。

- 使用 `cv2.boundingRect()` 找出物件的最小矩形框。

- 使用 `cv2.minEnclosingCircle()` 找出物件的最小圓形框。

- 使用 `cv2.minAreaRect()` 找出物件的最小旋轉矩形框。

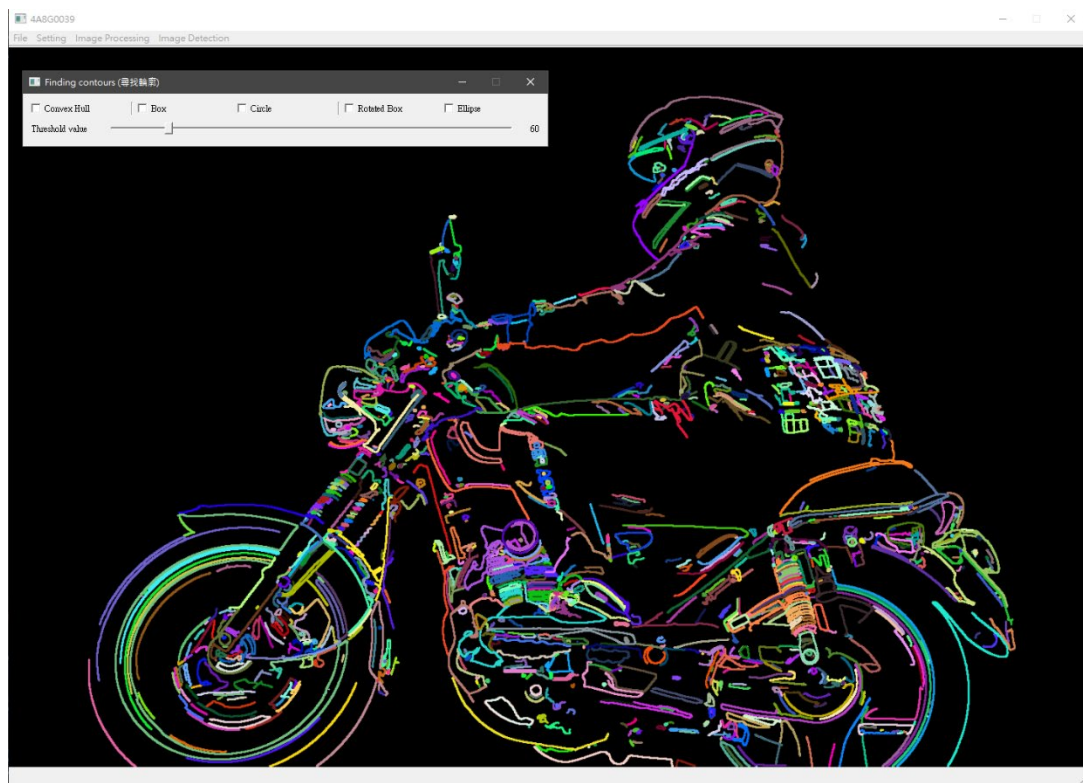
- 使用 `cv2.fitEllipse()` 找出物件的最小橢圓形框。

Threshold value : Canny 閾值

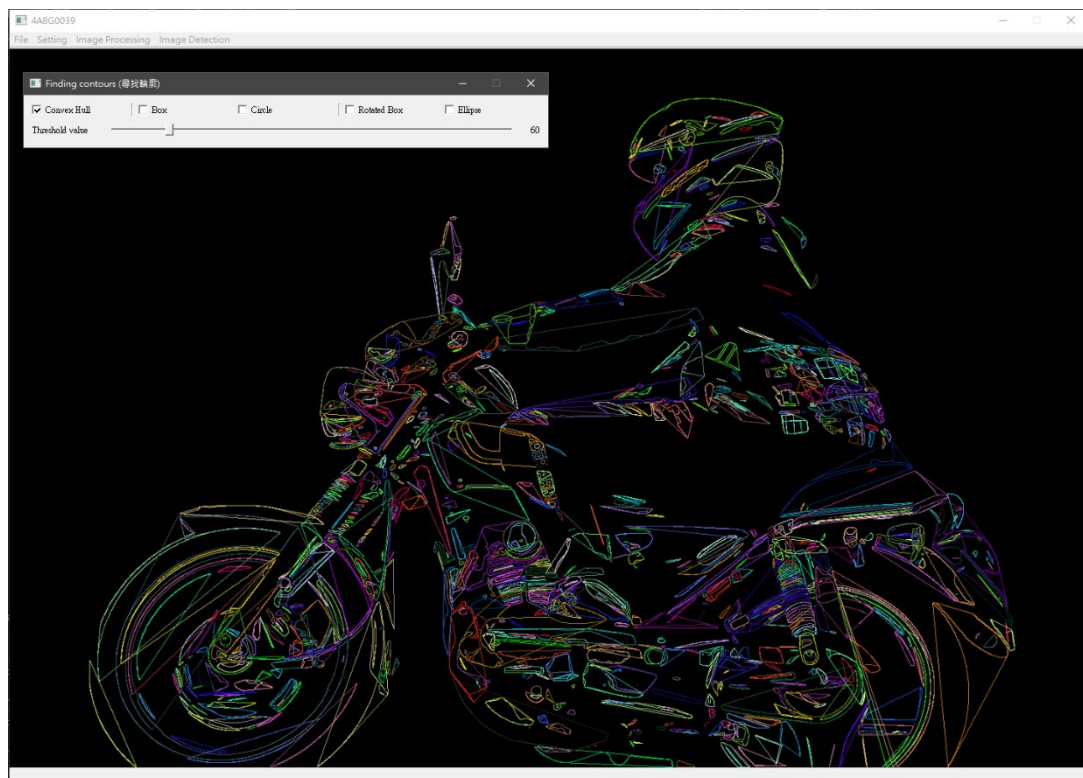




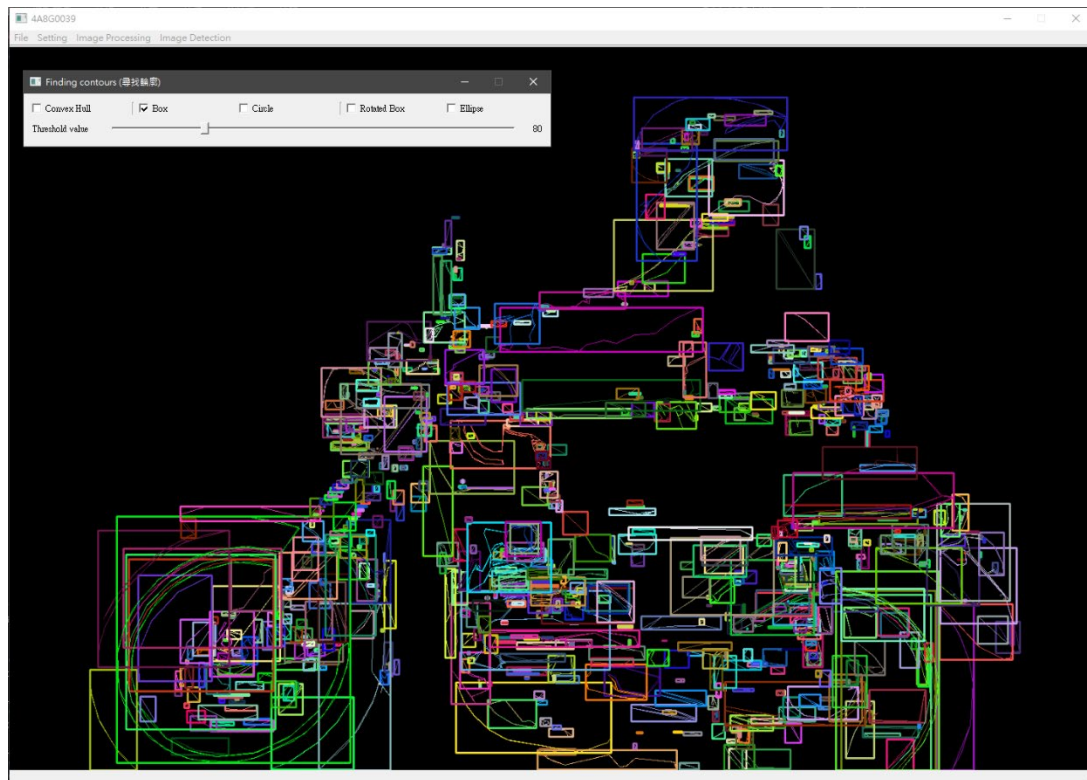
cv2.findContours :



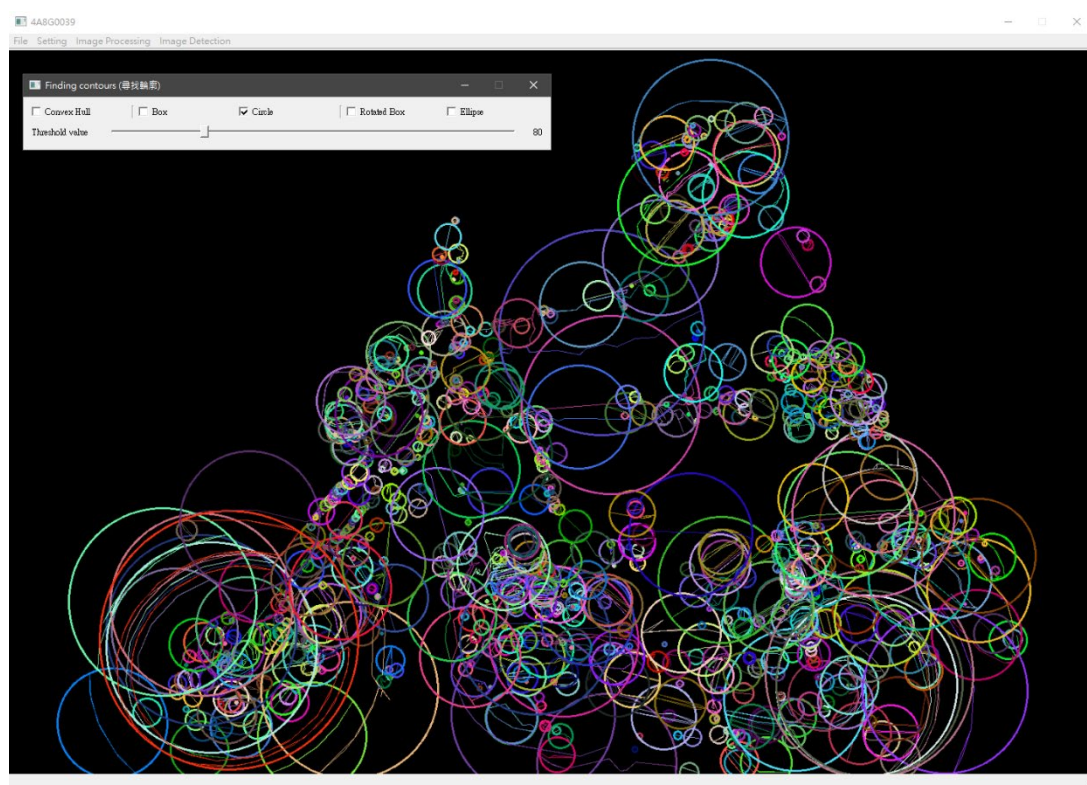
cv2.convexHull :



cv2.boundingRect :

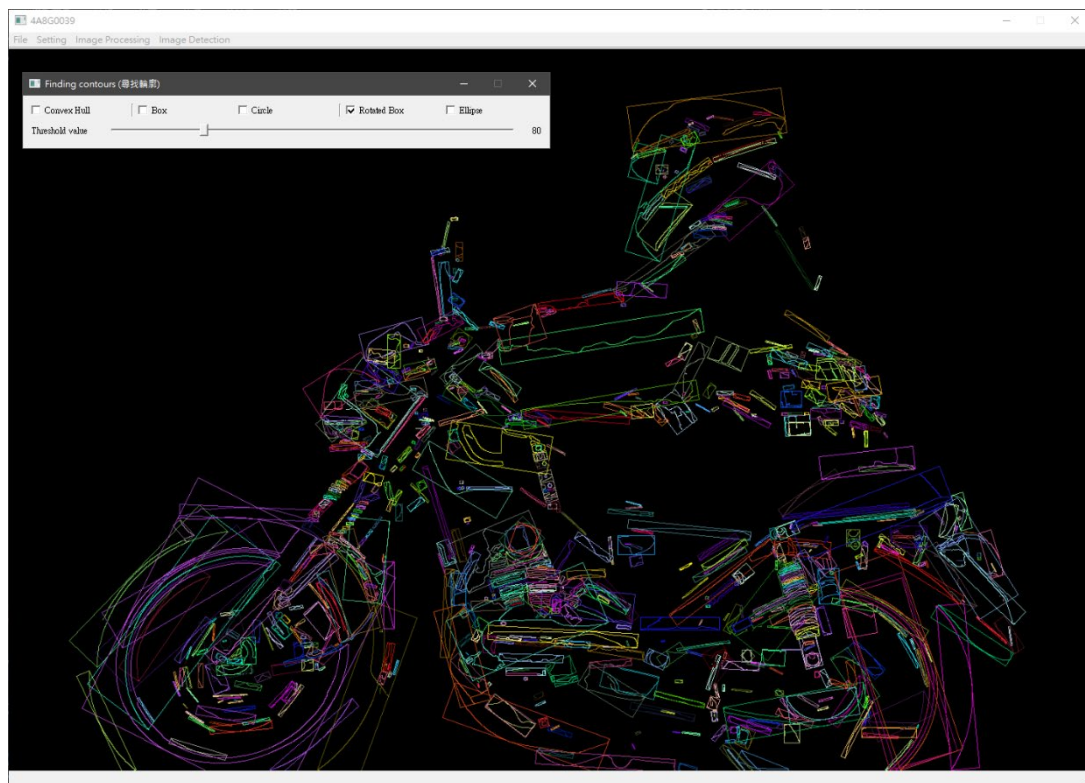


cv2.minEnclosingCircle :

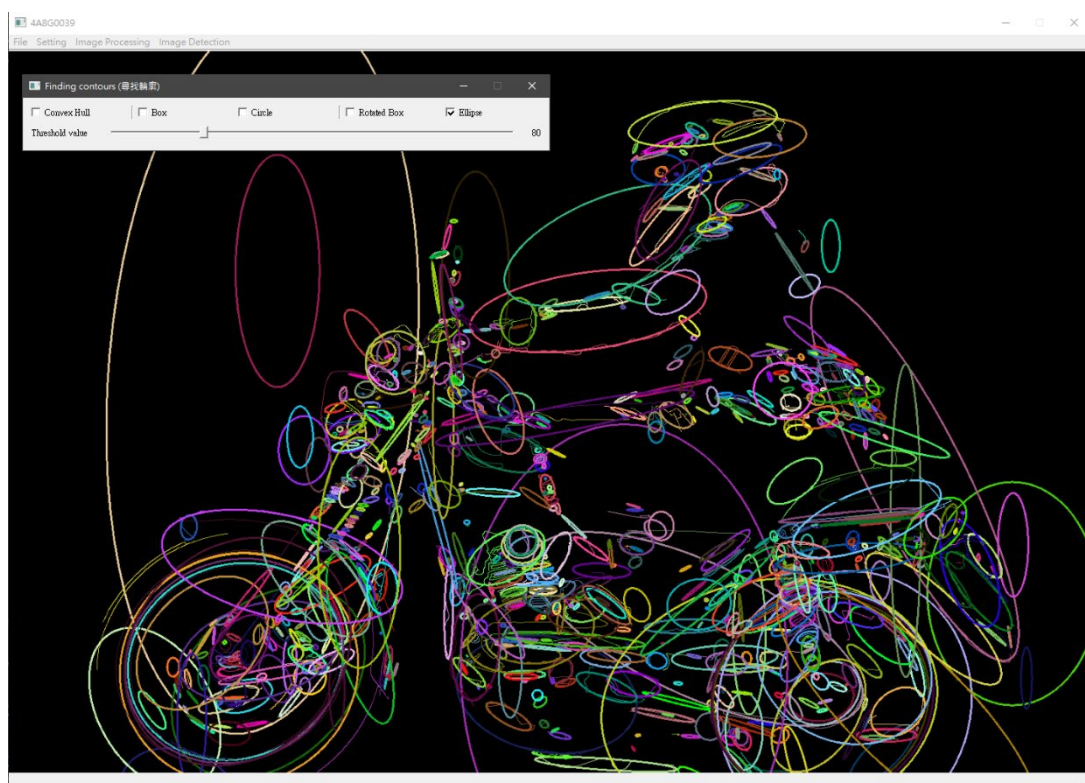




cv2.minAreaRect :



cv2.fitEllipse :



## Morphology Transformations (形態轉換) :

使用 `cv2.MORPH_ERODE` 實作腐蝕圖像。

使用 `cv2.MORPH_DILATE` 實作擴大圖像。

使用 `cv2.MORPH_OPEN` 實作腐蝕圖像後擴大圖像。

使用 `cv2.MORPH_CLOSE` 實作擴大圖像後腐蝕圖像。

使用 `cv2.MORPH_GRADIENT` 實作擴大圖像 - 腐蝕圖像。

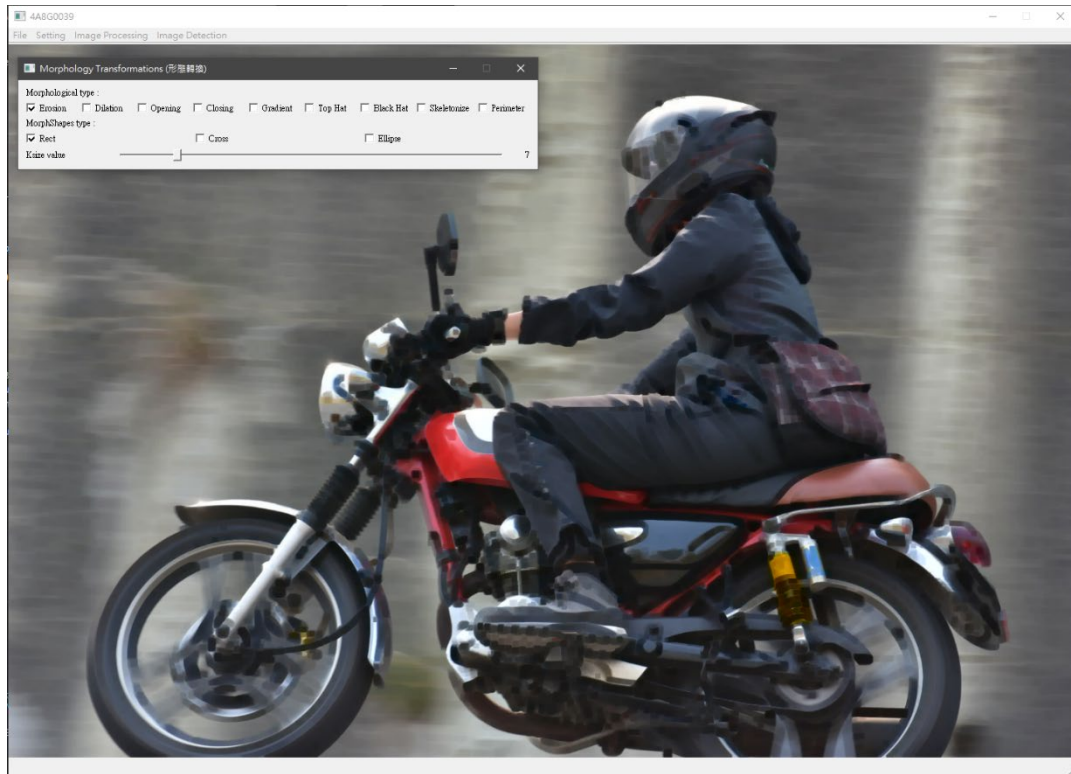
使用 `cv2.MORPH_TOPHAT` 實作原圖像 - `cv2.MORPH_OPEN`。

使用 `cv2.MORPH_BLACKHAT` 實作  
`cv2.MORPH_CLOSE` - 原圖像。

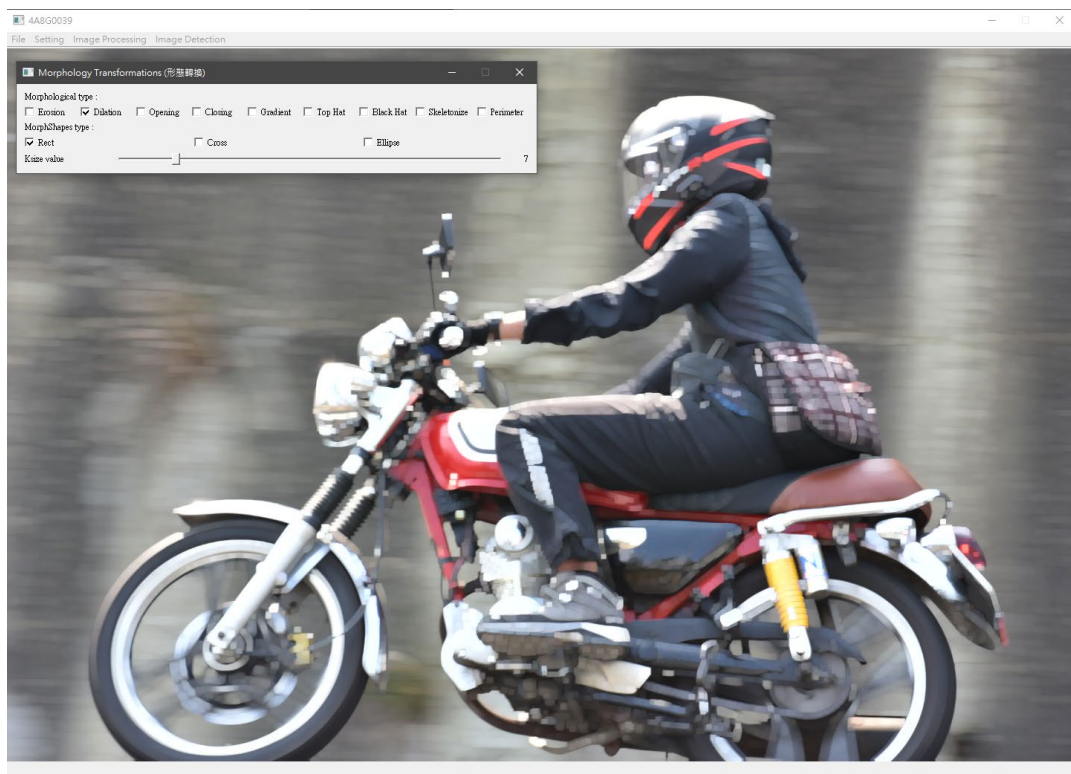
Ksize value : 模糊內核大小



cv2.MORPH\_ERODE :

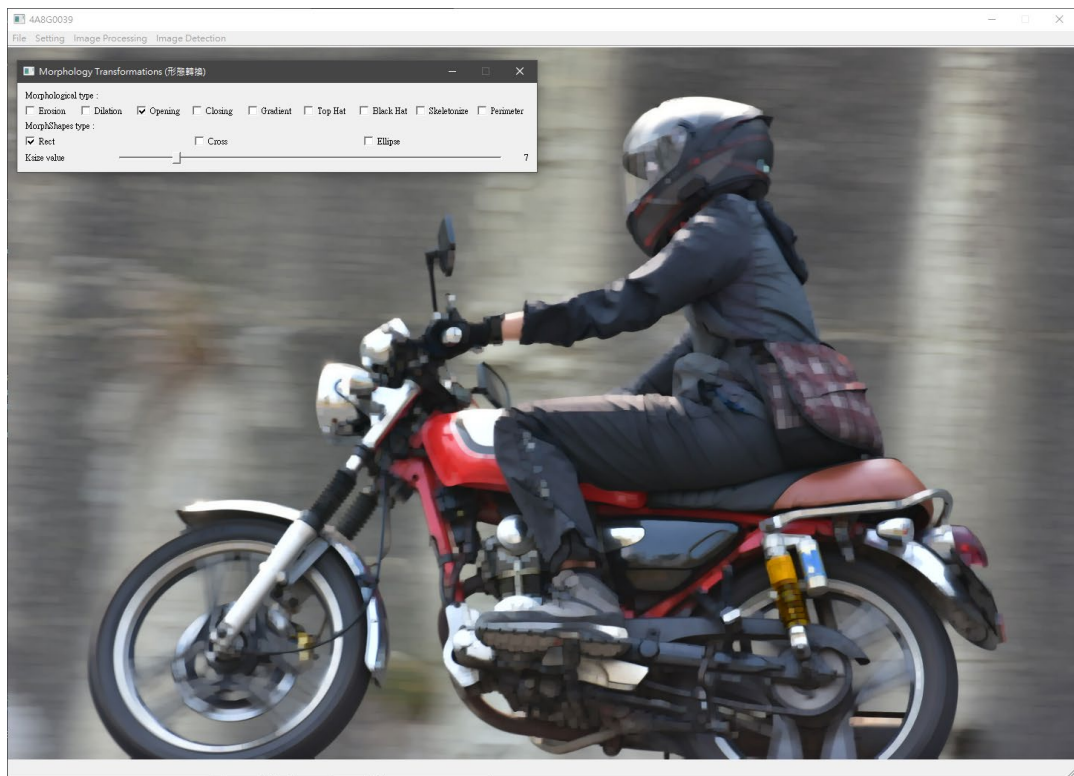


cv2.MORPH\_DILATE :

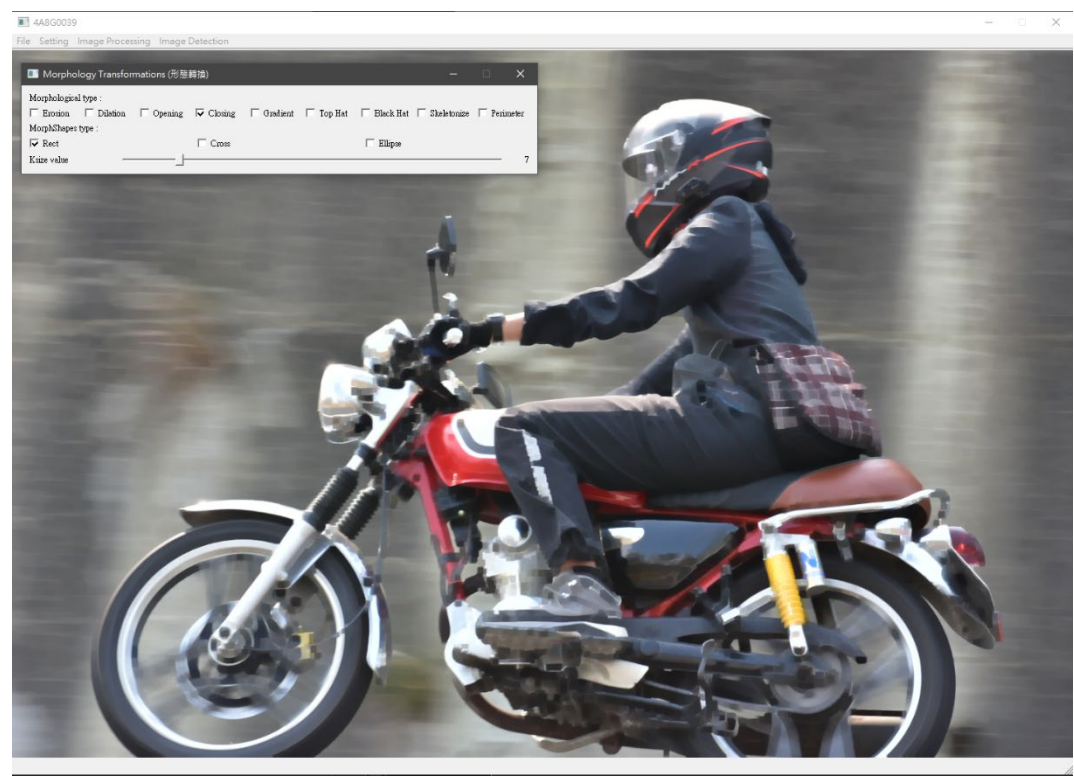




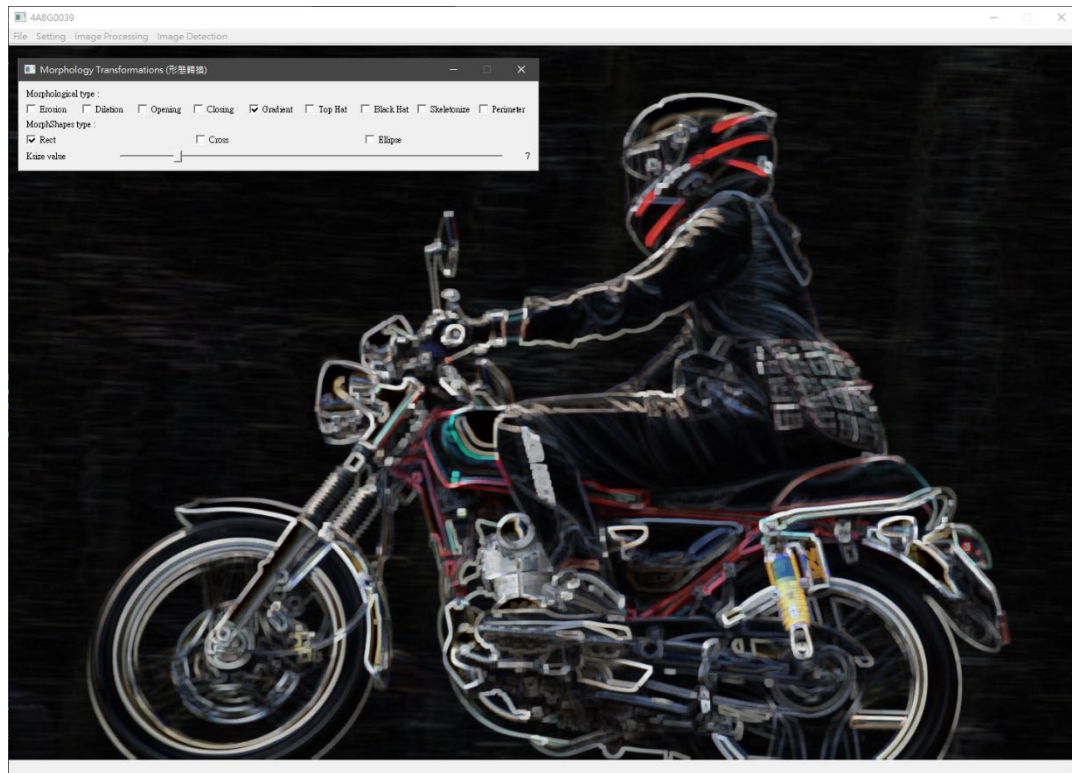
cv2.MORPH\_OPEN :



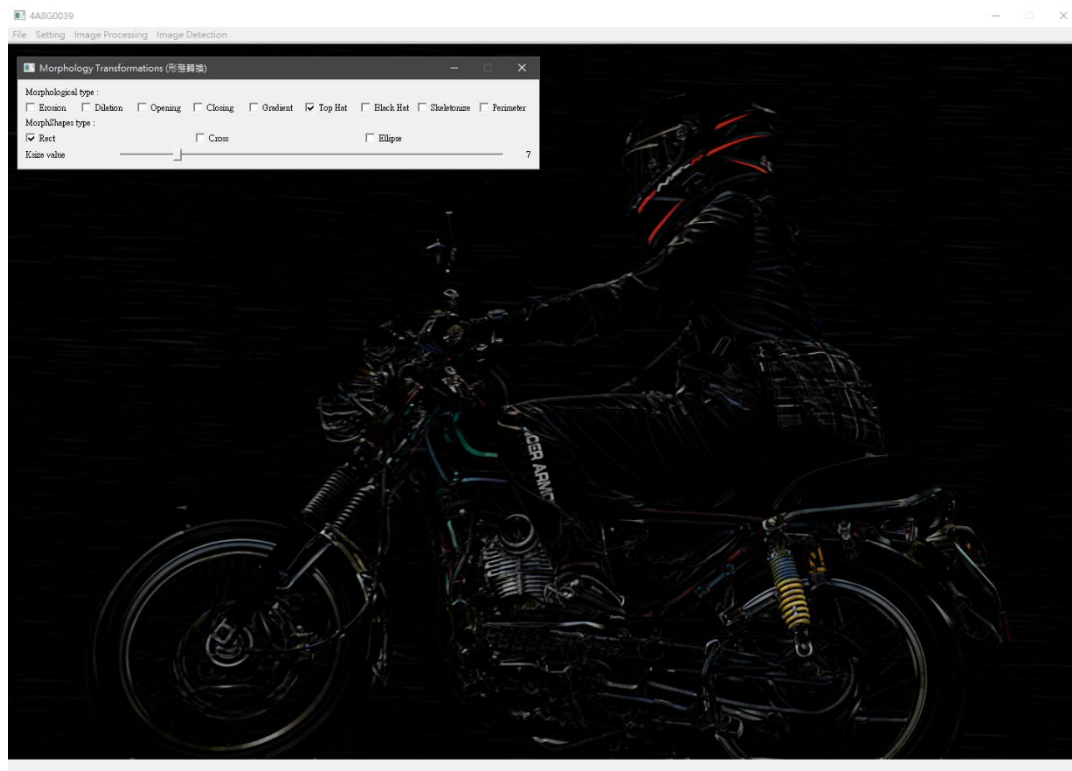
cv2.MORPH\_CLOSE



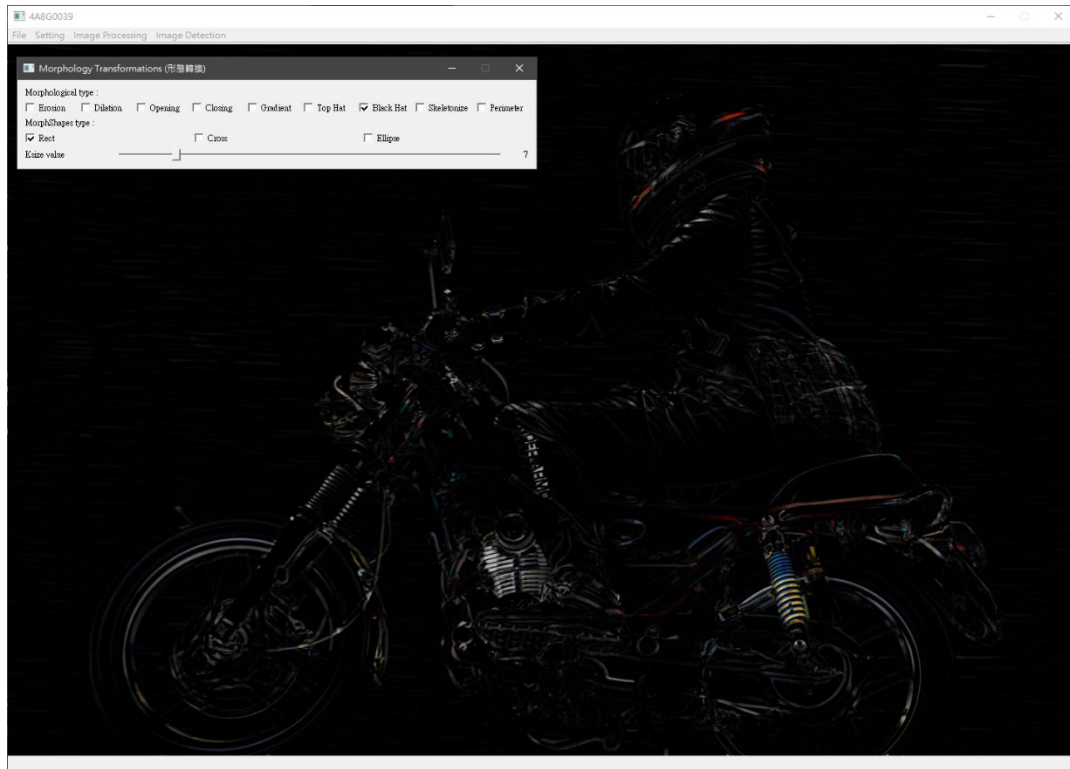
cv2. MORPH\_GRADIENT :



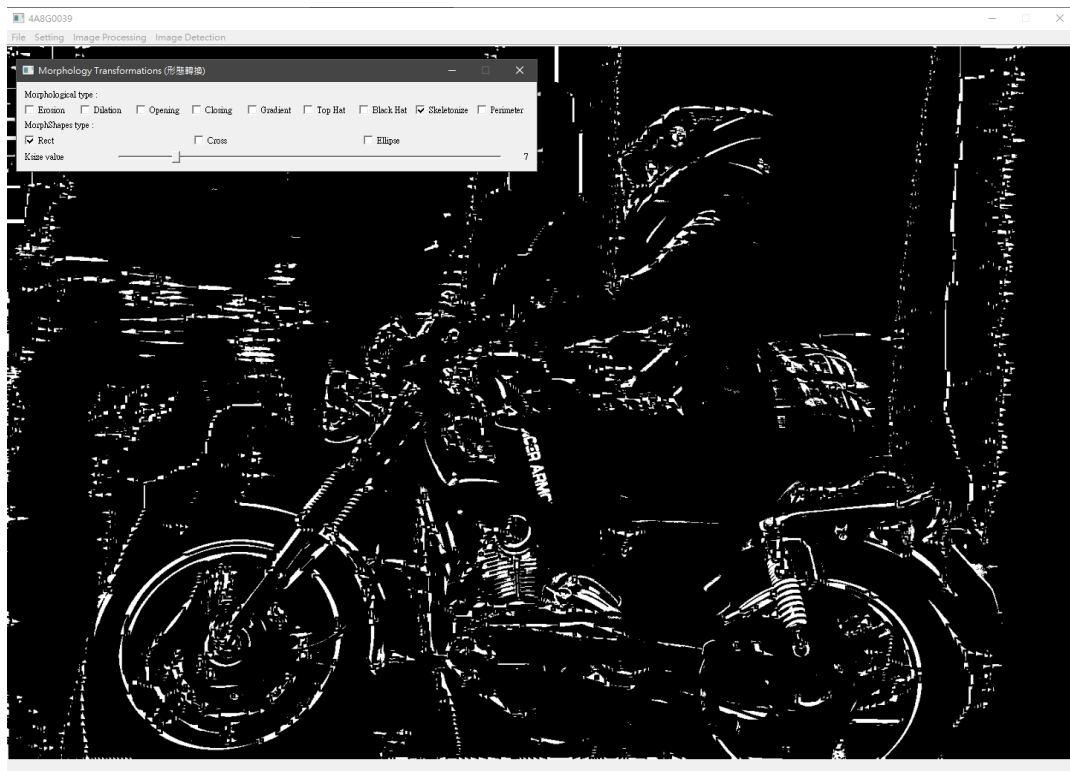
cv2.MORPH\_TOPHAT :



cv2.MORPH\_BLACKHAT :



Skeletonize :



Perimeter :

