**計劃名稱: LED頻率辨識與偵測**

**撰寫人: 陳鎮華**

**最後更新: 2019/10/12**

**目錄**

[系統簡介 3](#_Toc22484906)

[系統環境 3](#_Toc22484907)

[系統流程 3](#_Toc22484908)

[系統架構 4](#_Toc22484909)

[演算法分析 7](#_Toc22484910)

[相關連結 9](#_Toc22484911)

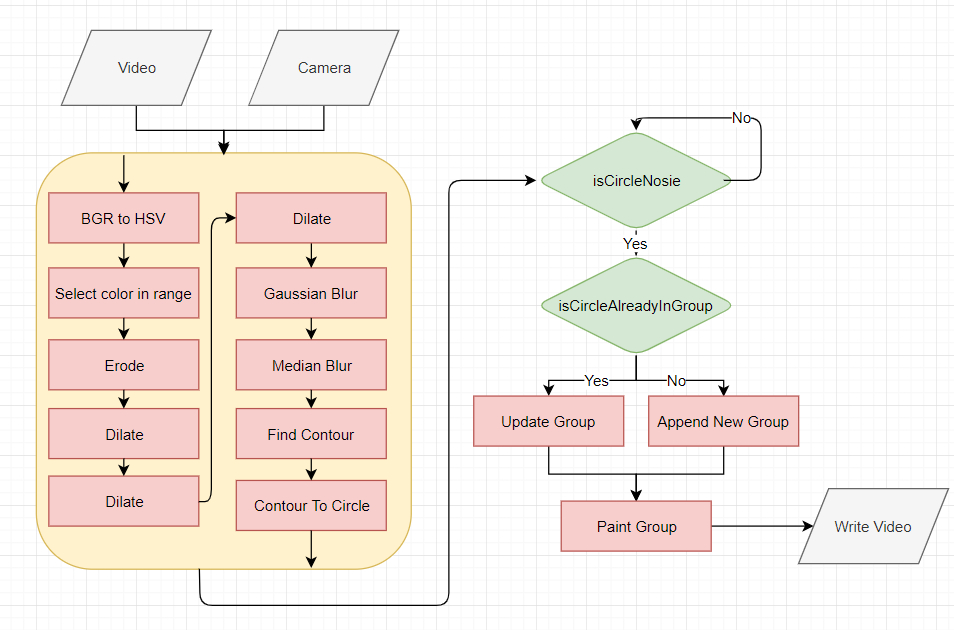
# 系統簡介

本程式的功能主要用於LED頻率之辨識，運用OpenCV影像辨識與處理技巧與Python程式語言進行撰寫。目的用於辨識魚體身上之LED燈光頻率，目前可辨識**紅**、**藍**、**綠**三種顏色之燈光及複數目標之辨識，並產生頻率資料之表格(.csv)及處理過後的影片資料(.mp4)。使用者可依需求透過使用者介面在程式執行時調整辨識之各參數(HSV範圍、Group範圍)，也可在程式碼頂端調整起始參數。目前演算法對於移動速度快、Group互相交疊以及顏色範圍不夠明確(紅色)之魚體辨識能力較弱。

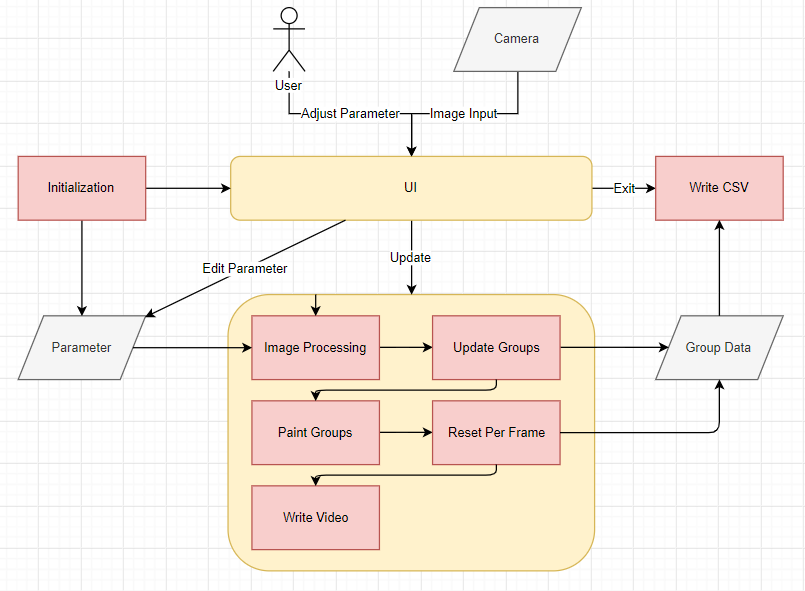
# 系統環境

* 語言: Python 3.6.7
* 函式庫: OpenCV 4.0.1
* 作業系統: Window 10

# 系統流程

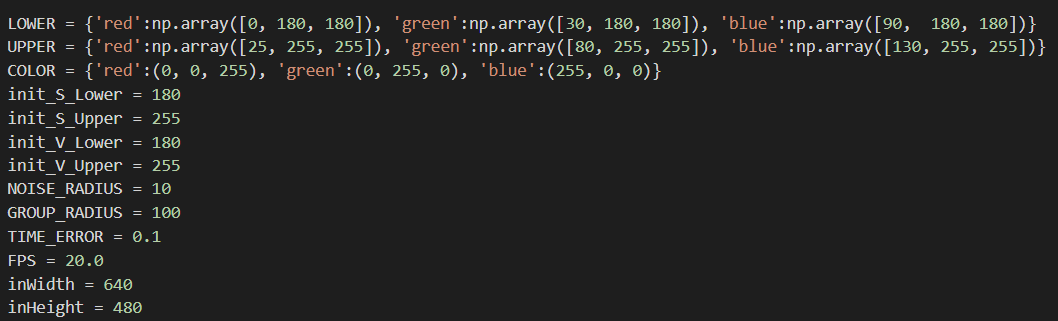


# 系統架構



1. **Initialization**

初始化UI與影像處理參數

相關函式: setTrackbar、程式碼最上方的參數調整

LOWER 影像HSV之下限

UPPER 影像HSV之上限

COLOR 繪製影像之BGR顏色

init\_S\_Lower/Upper 初始S之下限/上限

init\_V\_Lower/Upper 初始V之下限/上限

NOISE\_RADIUS 雜訊之半徑(小於此大小將會被判定為雜訊)

GROUP\_RADIUS Group之半徑(在此大小內將被視為同一Group)

TIME\_ERROR 時間容忍之誤差範圍

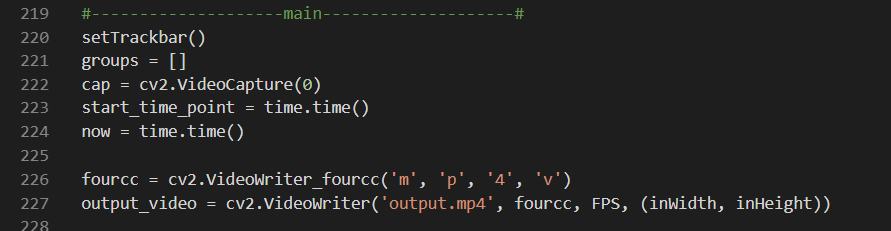
FPS 影片幀數

inWidth/inHeight 輸入影像之長寬

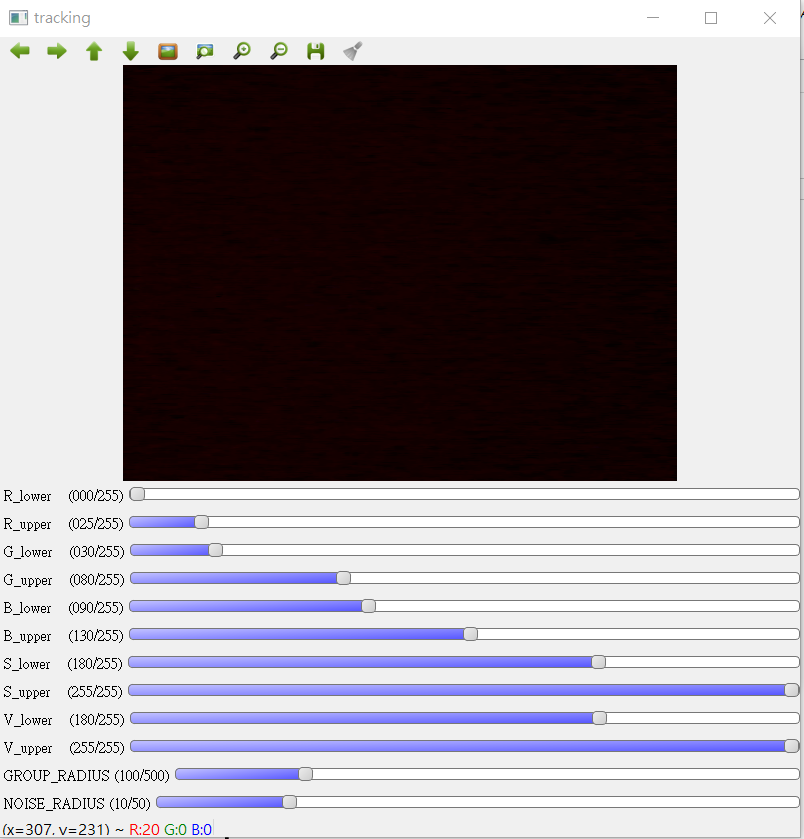
1. **Input**

Camera影像輸入可透過參數調整

相關函式: main部分的 cap



1. **UI**



可透過UI調整影像處理之參數以及處理過後之影像即時繪製

相關函式: modifyParameter

1. **Update**
   1. **Image Processing**

透過OpenCV進行影像處理，詳細演算法在演算法分析部分說明

相關函式: updateGroups, selectColor

* 1. **Update Groups**

處理完的影像資料與Groups進行比較並更新以及辨識出部分影像繪製

相關函式: updateGroups, contour2Group, matchGroup, contourInGroup, updateColorInGroup

* 1. **Paint Groups**

將每個Group繪製上影像

相關函式: paintGroups, isGroupLEDLightOn, calCenter

* 1. **Reset Per Frame**

確認Group內的LED是否持續亮著並調整重製

相關函式: resetGroups

* 1. **Write Video**

將處理完的影像寫成mp4檔

相關函式: writeVideo

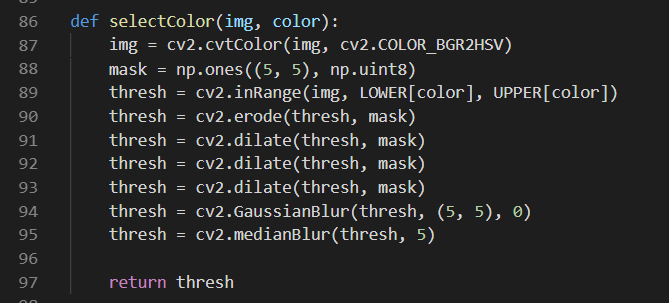
1. **Write CSV**

將Group內的資料寫成csv表格

相關函式: writeFile

# 演算法分析

1. **selectColor**



將影像按照以下順序進行處理

cv2.cvtColor 從BGR色彩空間轉為HSV色彩空間

cv2.inRange 將顏色參數之範圍內的影像濾出

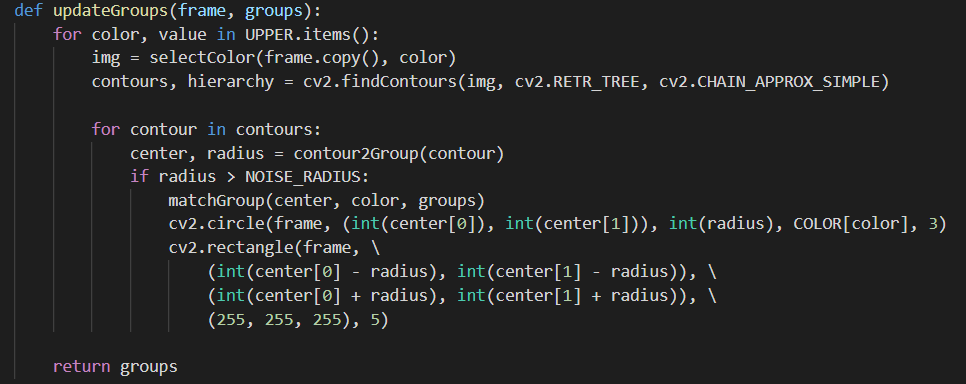
cv2.erode 影像處理之侵蝕運算(消除雜點)

cv2.dilate 影像處理之膨脹運算(將影像破洞補起)

cv2.GaussianBlur 影像處理之高斯模糊化(使影像較接近真實影像)

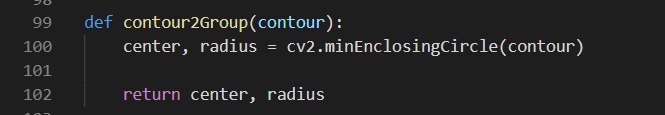
cv2.medianBlur 影像處理之中位數模糊化(使影像偏差降低)

1. **updateGroups**



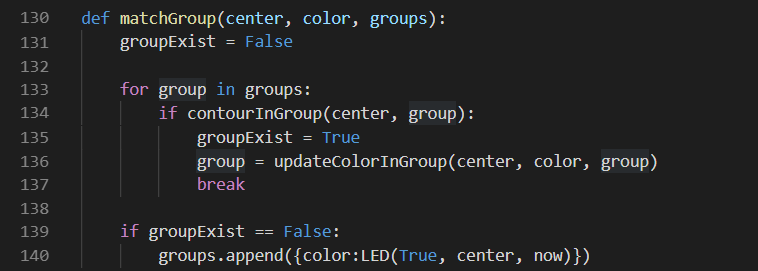
cv2.findContours 影像處理找出畫面中的凸包

for 迴圈將所有contour與group進行比較並更新與繪製

1. **contour2Group**

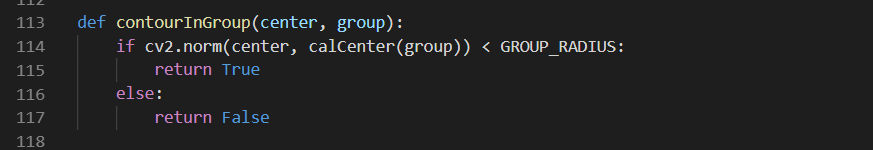
cv2.minEnclosingCircle 影像處理透過凸包找出近似之圓形

1. **matchGroup**



將所有各個center與group交叉比較、分類並決定更新或新增group

1. **contourInGroup**



cv2.norm 凸包與group計算距離決定是否歸類為同一group

# 相關連結

Github: <https://github.com/shizsun0609tw/LED_Recognizaion>

OpenCV: <https://reurl.cc/e5WnVx>

凸包: <https://zh.wikipedia.org/wiki/%E5%87%B8%E5%8C%85>

侵蝕與膨脹: <https://reurl.cc/72jnbb>

高斯模糊: <https://reurl.cc/K6XqOg>

中值模糊: <https://en.wikipedia.org/wiki/Median_filter>

Draw.io: <https://www.draw.io/>