

# 阿洲的程式教學

關於Qt、OpenCV、影像處理演算法

---

## 輪廓(findContours、drawContours)

當我們做物件辨識時，透過輪廓可得到特定物件的資訊，協助我們做判斷，OpenCV的findContours()函式可找到影像的輪廓，依實際需求調整參數輸入，而這邊的輪廓和Sobel這些找邊緣的處理不同，Sobel是將物件內部消除，只保留物件邊緣，findContours是在經過Sobel處理之後，將這個只有邊緣的影像，把各個邊緣點做分類，連結的邊緣點儲存在同個容器內，當我們找到輪廓後，可用drawContours()劃出輪廓線，檢查是否取得正確或適合的輪廓。

以下列出兩個多載使用的函式：

### OpenCV找輪廓

```
void findContours(InputOutputArray image, OutputArrayOfArrays contours,  
OutputArray hierarchy, int mode, int method, Point offset=Point())
```

```
void findContours(InputOutputArray image, OutputArrayOfArrays contours, int  
mode, int method, Point offset=Point())
```

- **image**：輸入圖，使用八位元單通道圖，所有非零的像素都會列入考慮，通常為二極化後的圖。
- **contours**：包含所有輪廓的容器(vector)，每個輪廓都是儲存點的容器(vector)，所以contours的資料結構為vector< vector>。
- **hierarchy**：可有可無的輸出向量，以階層的方式記錄所有輪廓。
- **mode**：取得輪廓的模式。
- **method**：儲存輪廓點的方法。

**mode**：取得輪廓的模式，有以下幾種可選擇：

- **CV\_RETR\_EXTERNAL**：只取最外層的輪廓。

- **CV\_RETR\_LIST**：取得所有輪廓，不建立階層(hierarchy)。
- **CV\_RETR\_CCOMP**：取得所有輪廓，儲存成兩層的階層，首階層為物件外圍，第二階層為內部空心部分的輪廓，如果更內部有其餘物件，包含於首階層。
- **CV\_RETR\_TREE**：取得所有輪廓，以全階層的方式儲存。

**method**：儲存輪廓點的方法，有以下幾種可選擇：

- **CV\_CHAIN\_APPROX\_NONE**：儲存所有輪廓點。
- **CV\_CHAIN\_APPROX\_SIMPLE**：對水平、垂直、對角線留下頭尾點，所以假如輪廓為一矩形，只儲存對角的四個頂點。

## OpenCV畫輪廓線

`void drawContours(InputOutputArray image, InputArrayOfArrays contours, int contourIdx, const Scalar& color, int thickness=1, int lineType=8, InputArray hierarchy=noArray(), int maxLevel=INT_MAX, Point offset=Point())`

- **image**：輸入輸出圖，會將輪廓畫在此影像上。
- **contours**：包含所有輪廓的容器(vector)，也就是**findContours()**所找到的 contours。
- **contourIdx**：指定畫某個輪廓。
- **color**：繪製的顏色。
- **lineType**：繪製的線條型態。
- **hierarchy**：輪廓階層，也就是**findContours()**所找到的**hierarchy**。
- **maxLevel**：最大階層的輪廓，可以指定想要畫的輪廓，有輸入**hierarchy**時才會考慮，輸入的值代表繪製的層數。

**maxLevel**：

- **0**：繪製指定階層的輪廓。
- **1**：繪製指定階層的輪廓，和他的一階子階層。
- **2**：繪製指定階層的輪廓，和他的一階、二階子階層。
- 剩下數字依此類推。

以下程式碼我們先用**Canny()**標出物體的邊緣，接著呼叫**findContours()**將邊緣歸類為

輪廓，呼叫drawContours()讓我們確認找到哪些輪廓：

```
#include <stdio>
#include <opencv2/opencv.hpp>
using namespace cv;

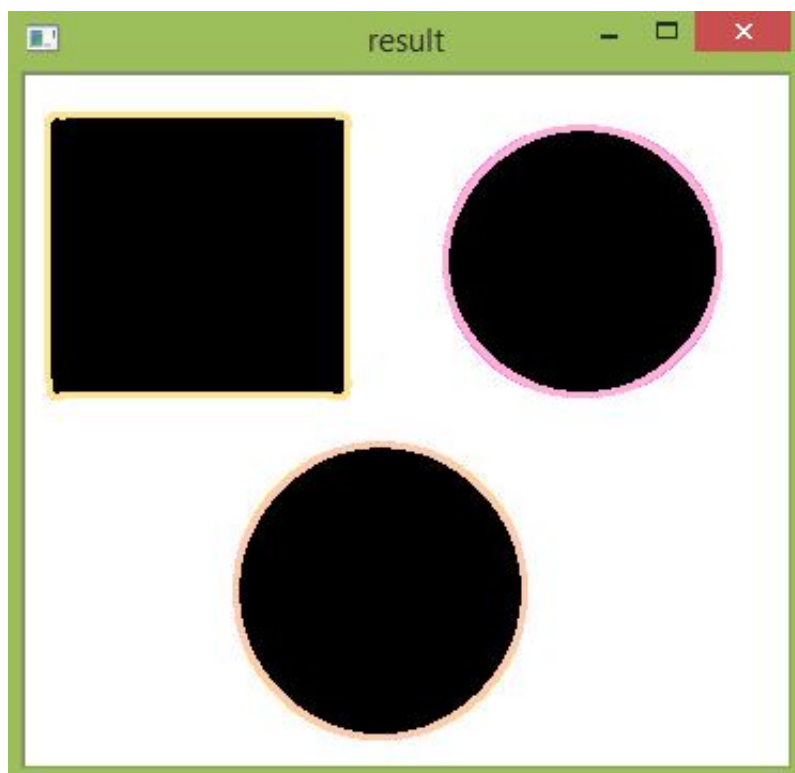
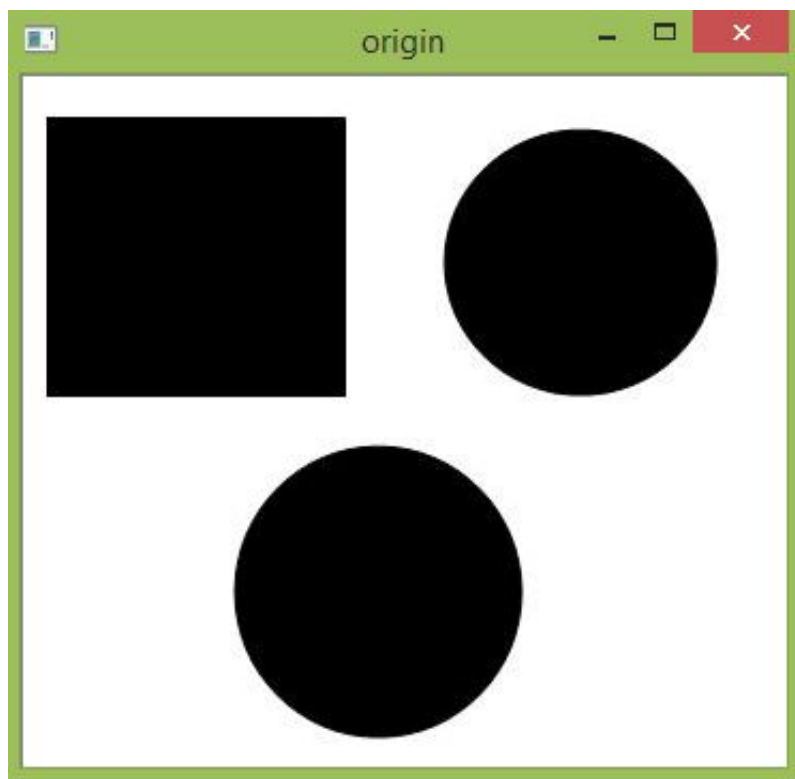
int main(){
    Mat src = imread("input.jpg", CV_LOAD_IMAGE_COLOR);
    Mat src_gray = imread("input.jpg", CV_LOAD_IMAGE_GRAYSCALE);
    Mat contoursImg = src.clone();

    Mat edge;
    blur(src_gray, src_gray, Size(3,3));
    Canny(src_gray, edge, 50, 150, 3);

    vector<vector<Point>> contours;
    vector<Vec4i> hierarchy;
    RNG rng(12345);
    findContours(edge, contours, hierarchy, CV_RETR_TREE,
CV_CHAIN_APPROX_NONE);
    for(int i = 0; i<contours.size(); i++){
        Scalar color = Scalar( rng.uniform(0, 255), rng.uniform(0, 255),
255);
        drawContours(contoursImg, contours, i, color, 2, 8, hierarchy);
    }

    imshow("origin", src);
    imshow("result", contoursImg);
    waitKey(0);

    return 0;
}
```



[回到首頁](#)

[回到OpenCV教學](#)

參考資料：

[OpenCV 教程](#)

📅 2015-11-30    👤 阿宅    📁 OpenCV, 輪廓    🔑 drawContours, findContours, 輪廓

**0 Comments**    猴子遇到0與1! 程式學習筆記

**1** Login ▾

♥ Recommend

🔗 Share

Sort by Best ▾



Start the discussion...

Be the first to comment.

**ALSO ON** 猴子遇到0與1! 程式學習筆記

### 文件對話框(QFileDialog)

1 comment • 6 months ago

楊政穎 — dialog.cpp 裡面的 QString s  
=  
QFileDialog::getOpenFileName(this, tr

### Qt主窗口(Top Level Window)

1 comment • 6 months ago

mike — 喔喔

✉ Subscribe

🗣 Add Disqus to your site Add Disqus Add

🔒 Privacy

自豪的採用 WordPress