

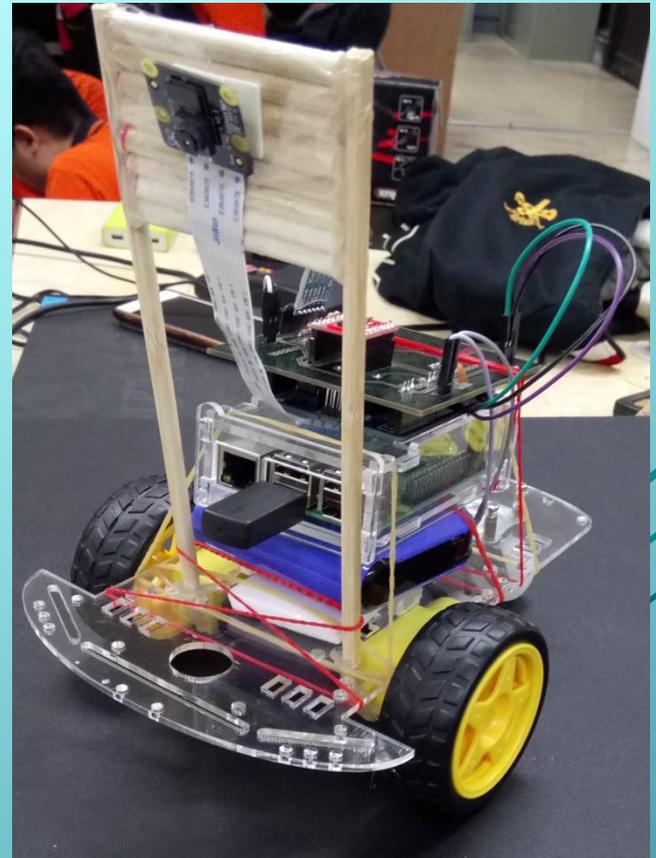
自動駕駛 Self-driving Car

指導老師：林家德 老師

組長：陳進銘

組員：林三文
陳柏儒
謝禎峻

報告日期：107/01/09 (二)



簡報大綱

一、基本介紹

- 1-1. 前言
- 1-2. 成員介紹
- 1-3. 甘特圖
- 1-4. 系統方塊圖

二、硬體介紹

- 2-1. TB6612FNG
- 2-2. RaspberryPi 3

三、原理介紹

- 3-1. 高斯濾波
- 3-2. OpenCV辨識

四、實驗過程及設計

五、程式介紹-樹莓派

- 5-1. 主程式
- 5-2. 副程式
- 5-3. 程式執行結果

六、程式介紹-Arduino

七、成果影片

八、結論與建議

一、基本介紹(1)前言

1. 研究動機：

台灣交通事故頻繁，但是在生活中，交通是必須的，交通安全也顯得特別重要。

2. 研究目的：

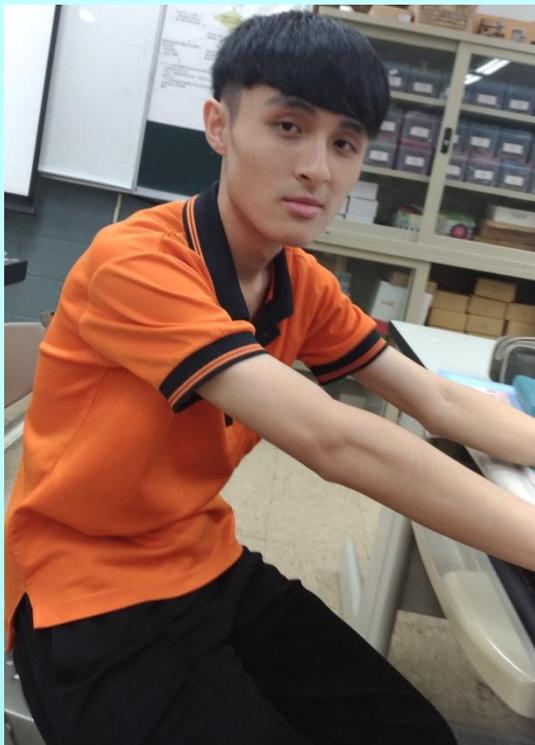
模擬自動駕駛之工作，使我們能了解自動駕駛的工作原理，並從中學習樹莓派3和複習Arduino。

(特斯拉)



一、基本介紹(2)成員介紹

陳進銘(26號)



負責工作：

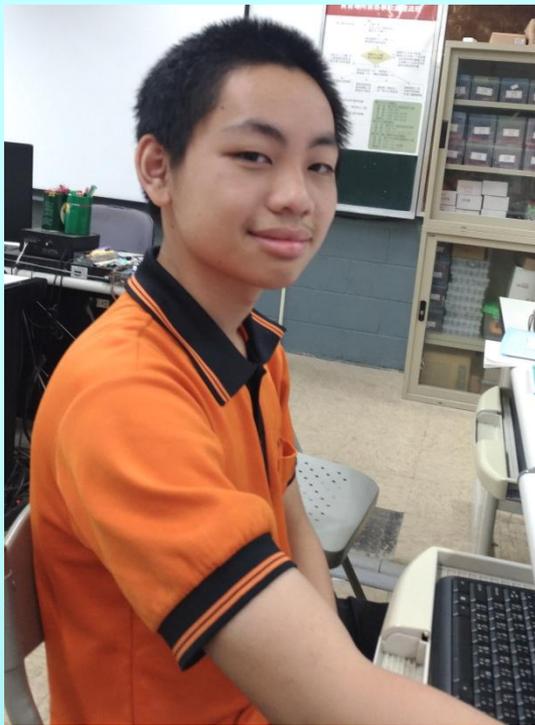
文書處理、
簡報製作、
口頭報告

經歷：

- (1)工業電子丙級技術士
- (2)105學年擔任輔導股長
- (3)106學年擔任副班長

一、基本介紹(2)成員介紹

林三文(9號)



負責工作：

Arduino程式撰寫、
協助樹莓派程式撰寫、
口頭報告

經歷：

- (1)工業電子丙級技術士
- (2)105學年擔任外掃衛生股長
- (3)105校內技藝競賽工業電子第四名
- (4)106學年擔任數學小老師
- (5)2017人工智慧鼠家居機器人國內第三名

一、基本介紹(2)成員介紹

陳柏儒(25號)



負責工作：

樹莓派程式設計、
協助硬體相關設計、
口頭報告

經歷：

(1)工業電子丙級技術士

一、基本介紹(2)成員介紹

謝禎峻(38號)



負責工作：

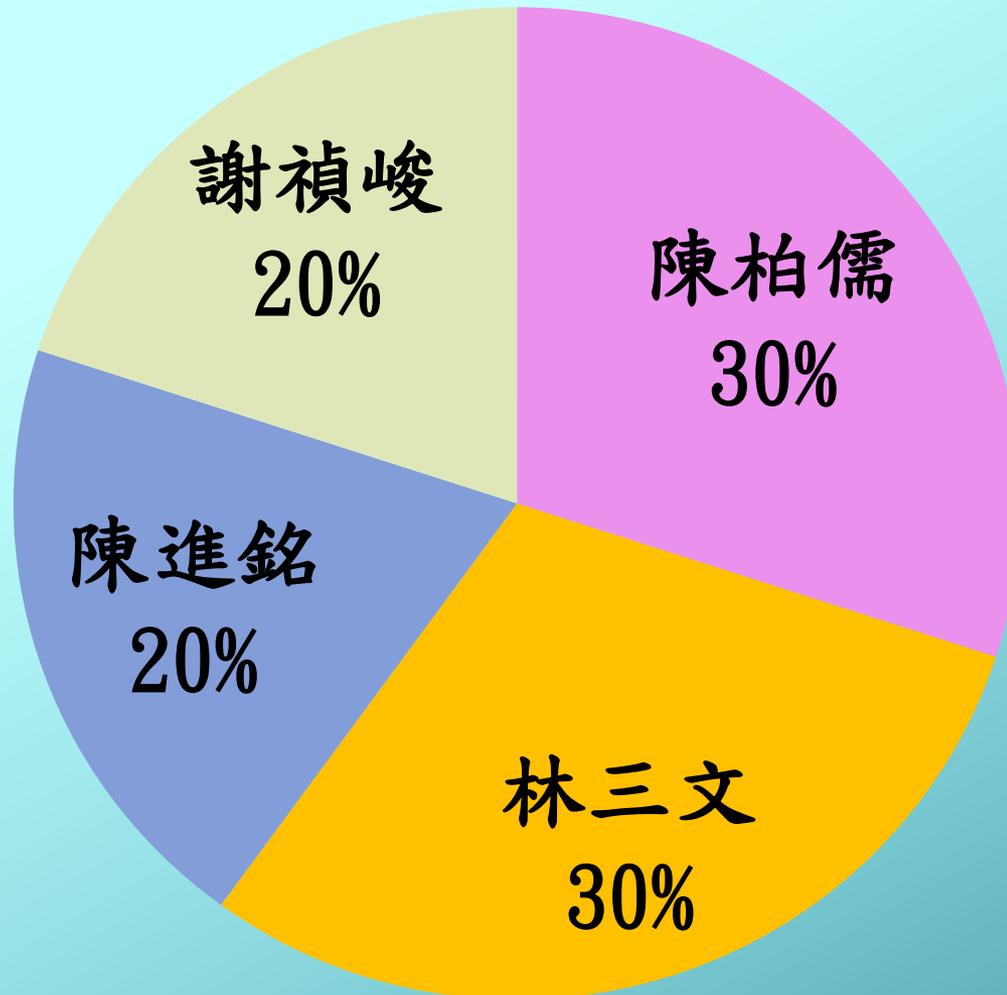
硬體相關設計、
焊接零件、
口頭報告

經歷：

- (1)工業電子丙級技術士
- (2)105學年擔任內掃衛生股長
- (3)106學年擔任總務股長

一、基本介紹(2)成員介紹

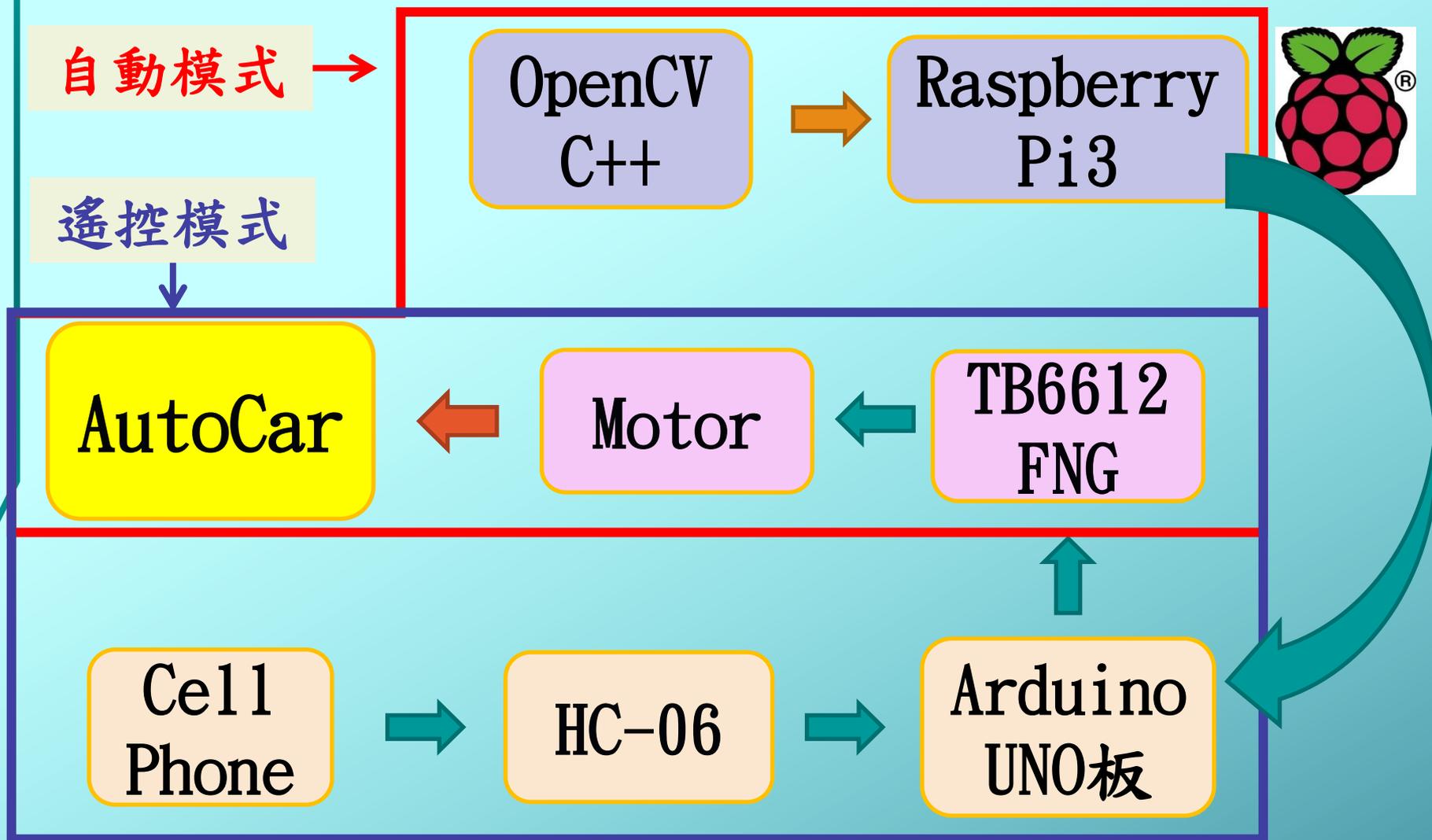
組員貢獻度



一、基本介紹(3)甘特圖

週次	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	負責成員
工作項目																			
資料蒐集	■	■	■																全體組員
理論探討	■	■	■	■															全體組員
專題準備			■	■															全體組員
機構規劃				■	■														儒、謝
機構製作				■	■														儒、謝
硬體規劃			■	■															儒、謝
硬體製作及測試				■	■														儒、謝
軟體規劃			■	■	■	■	■	■	■	■									三、銘
軟體規畫及測試									■	■	■	■							三、銘
整體測試															■	■	■		三、銘
製作印刷電路板												■	■	■					儒、謝
報告撰寫					■	■	■	■	■	■	■	■	■	■	■	■	■		全體組員
口頭報告						■			■		■		■			■		■	全體組員
預定進度	5	10	15	20	25	30	35	40	45	50	55	60	65	70	75	80	100		累積百分比%

一、基本介紹(4)系統方塊圖

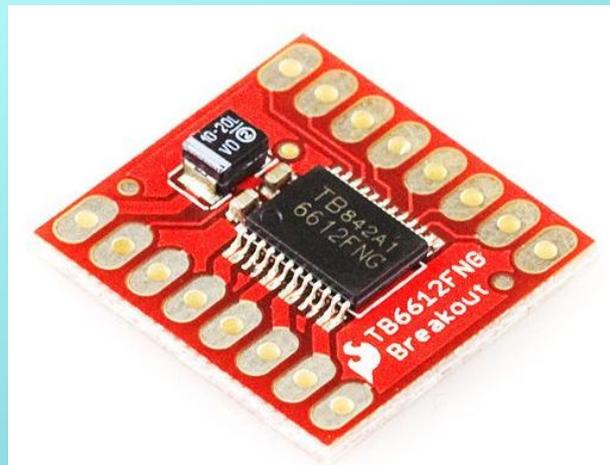


二、硬體介紹(1) TB6612FNG

1. 簡介

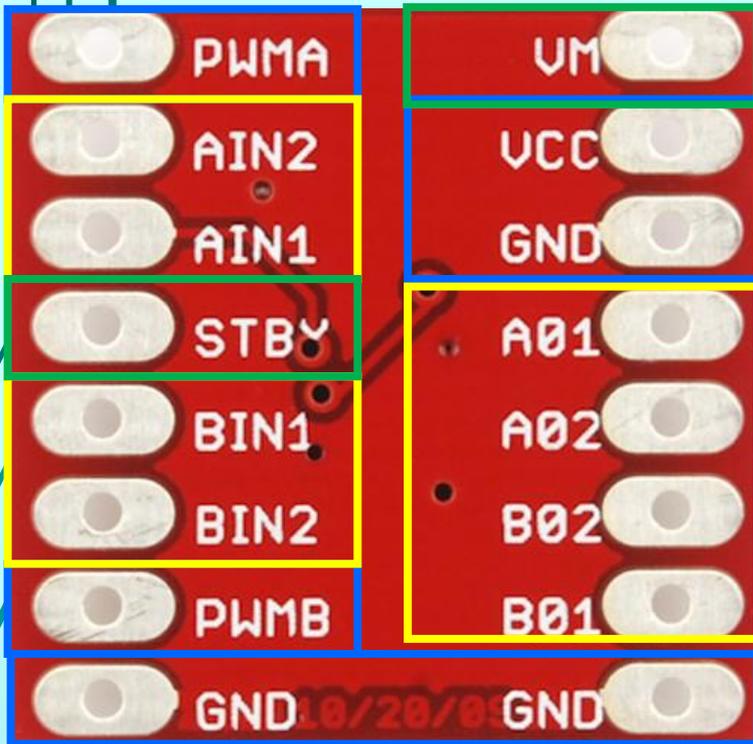
(1) TB6612FNG是東芝生產的馬達驅動與控制IC。

(2)內部包含兩組H橋式電路，可驅動和控制兩個小型直流馬達。



二、硬體介紹(1) TB6612FNG

2. 接腳介紹



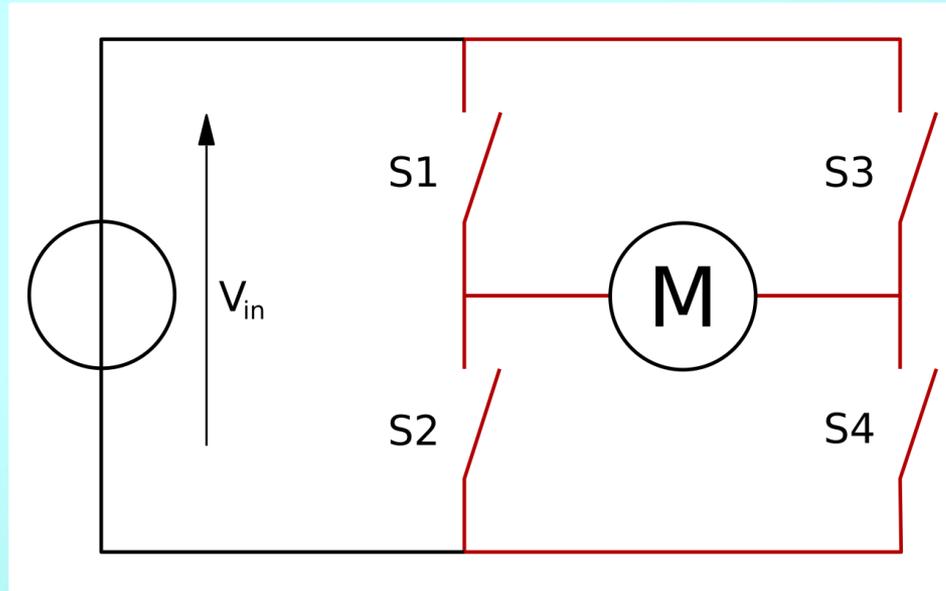
(1) **PWMA, PWMB** : 控制馬達轉速

(2) **A01, A02, B01, B02** : 輸出控制馬達A正反轉
BIN1, BIN2 : 控制馬達B正反轉

(3) **STBY** 馬達驅動源高電位動作

二、硬體介紹(1) TB6612FNG

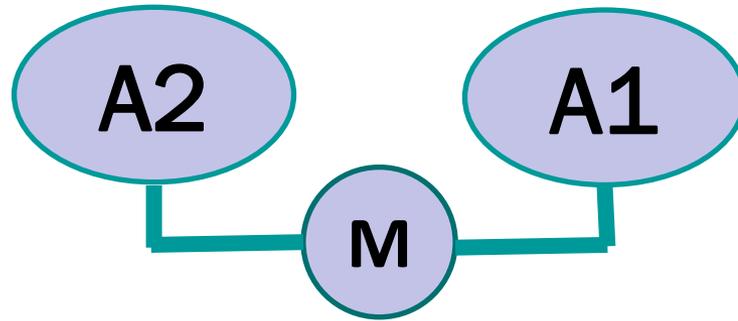
3. H橋介紹



S1	S2	S3	S4	狀態
0	1	1	0	馬達順向轉動
1	0	0	1	馬達反向轉動
0	0	0	0	馬達停止

二、硬體介紹(1) TB6612FNG

4. 控制模式 (馬達A)



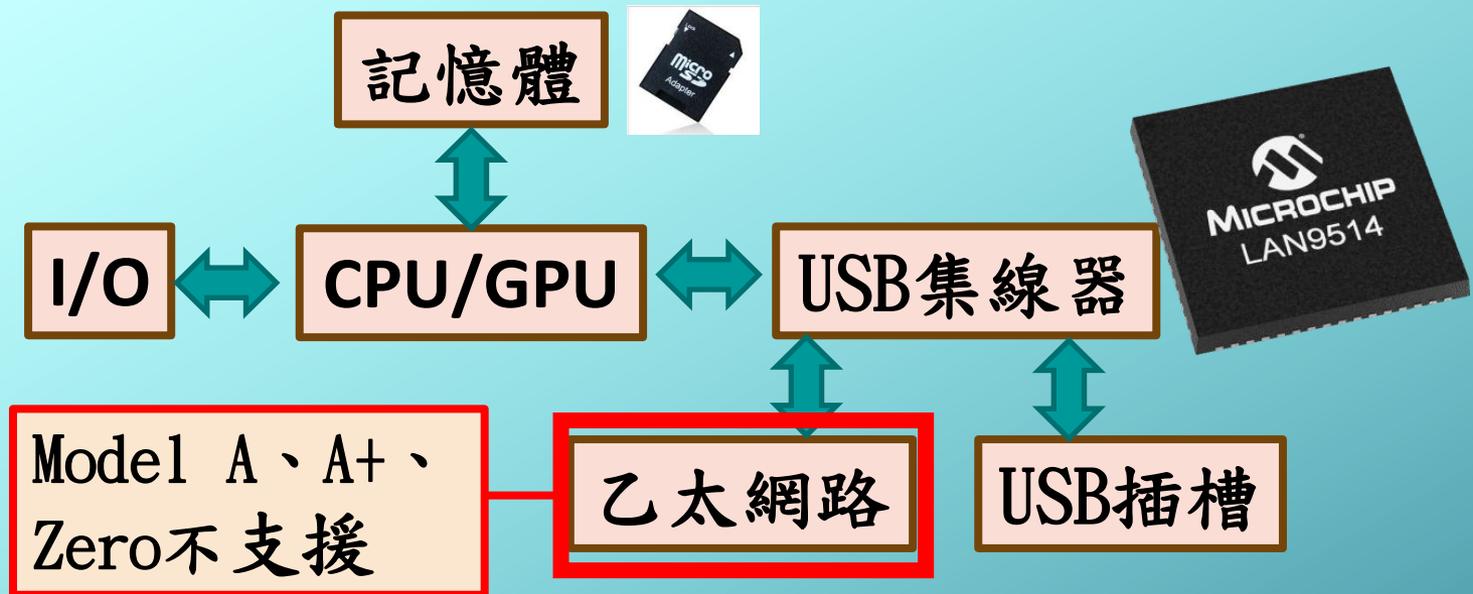
輸入			輸出		模式
A1	A2	PWMA	A01	A02	
0	0	1	高阻抗	高阻抗	停止
0	1	1	L	H	反轉
1	0	1	H	L	正轉
1	1	1	L	L	煞車

二、硬體介紹(2) RaspberryPi 3

1. 介紹:

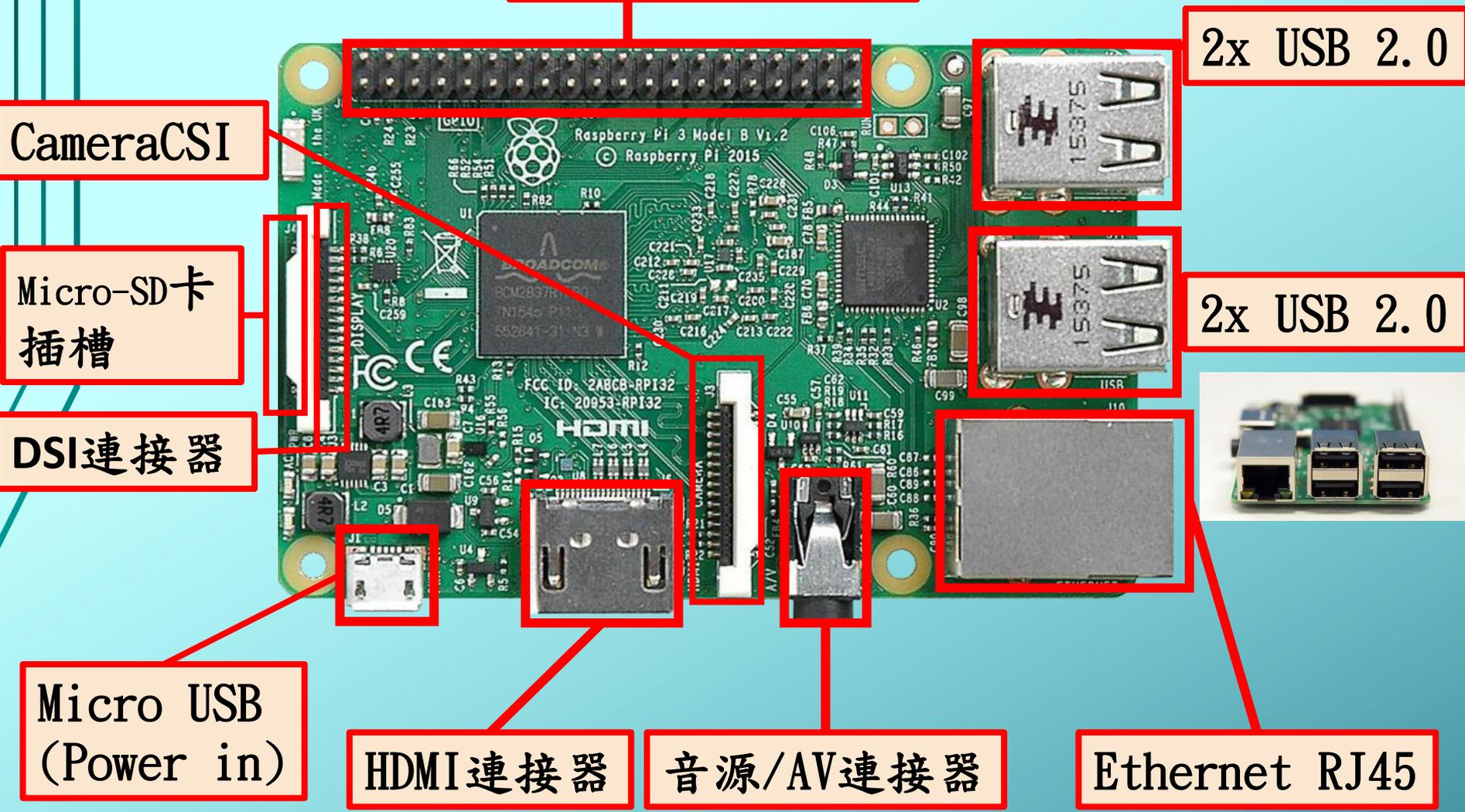
是一款基於Linux的單板機電腦，且具有ARM架構處理器。

2. 硬體的基本結構圖:



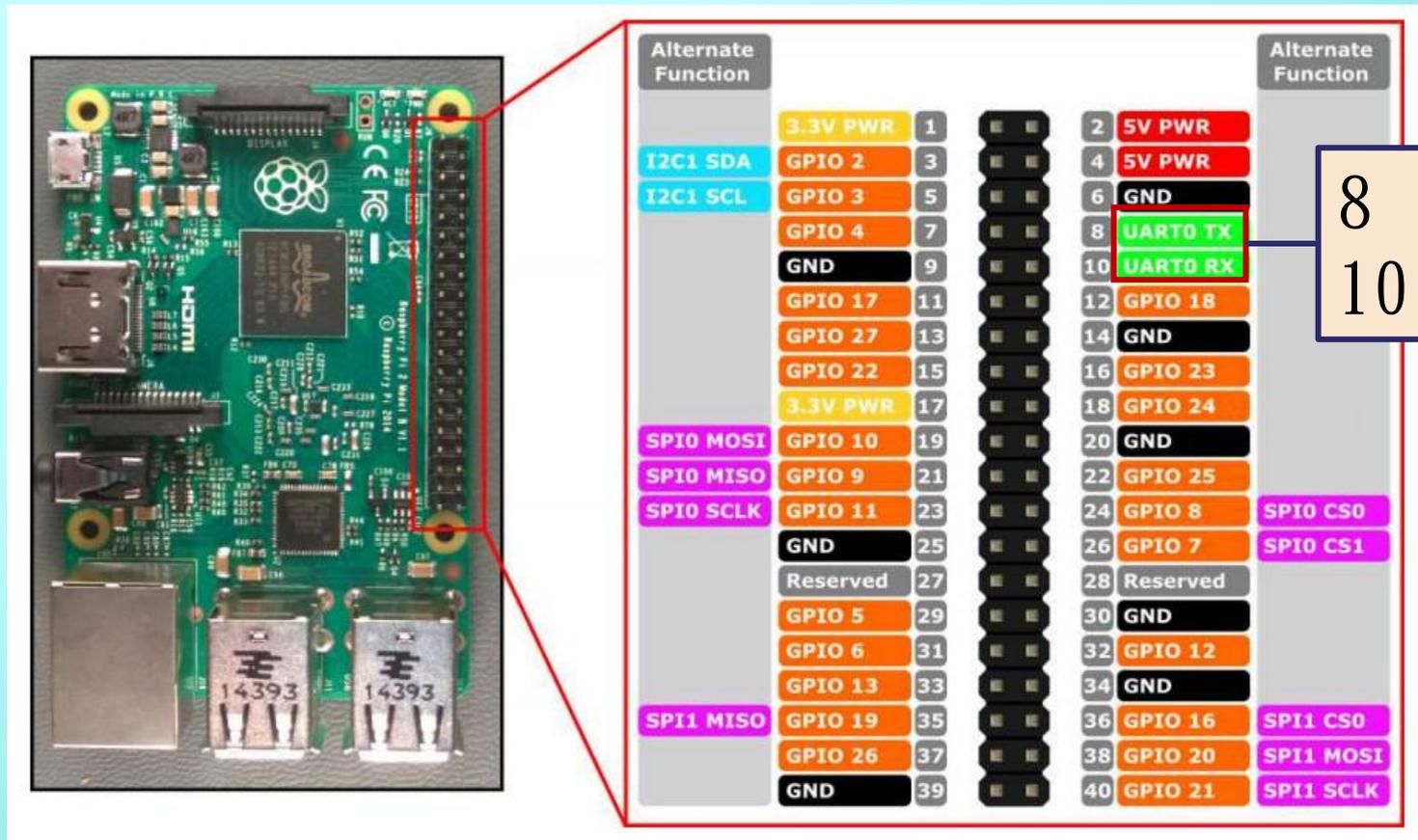
二、硬體介紹(2) RaspberryPi 3

3. 硬體配置：(40pin)GPIO接腳



二、硬體介紹(2) RaspberryPi 3

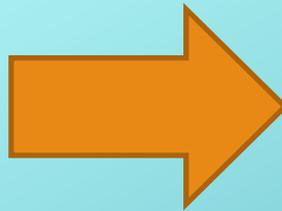
4. GPIO接腳



三、原理介紹(1) 高斯濾波

1. 介紹

即低通濾波，對整幅圖像進行加權平均的過程，降低影像的雜訊成分。

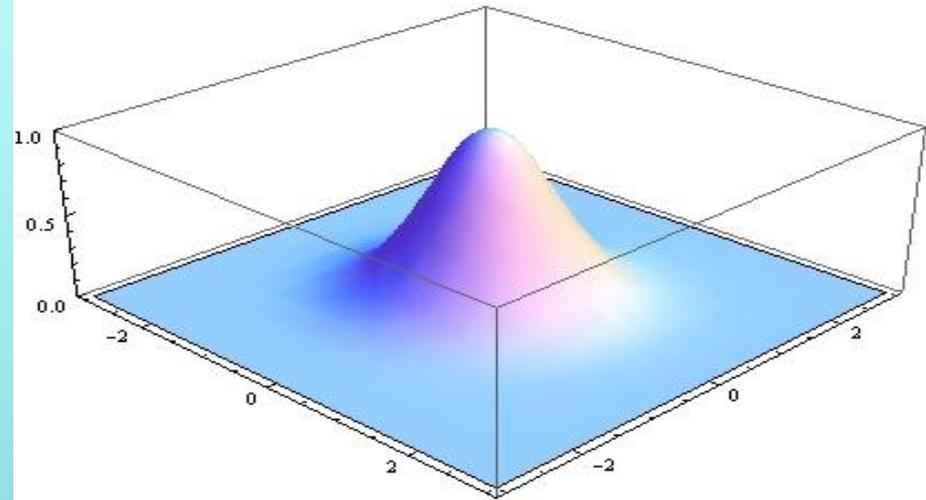


三、原理介紹(1) 高斯濾波

2. 高斯濾波器:

$$K = \frac{1}{159}$$

2	4	5	4	2
4	9	12	9	4
5	12	15	12	5
4	9	12	9	4
2	4	5	4	2



三、原理介紹(1) 高斯濾波

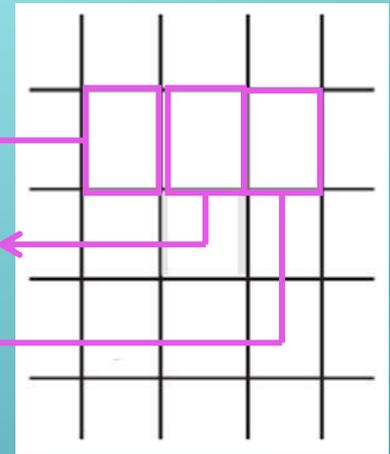
3. 模板運算:

濾波器

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

	a	b	c	
	d	e	f	
	g	h	i	

$$\begin{aligned} & 1/9(a*1+b*1+e*1+d*1) \\ & 1/9(a*1+b*1+c*1+f*1+e*1+d*1) \\ & 1/9(b*1+c*1+f*1+e*1) \end{aligned}$$



三、原理介紹(2) OpenCV辨識

1. 介紹：

OpenCV全名(Open Source Computer Vision Library)

(1)是一種開放原始碼的電腦視覺程式庫

(2)用C++語言編寫，它的主要介面也是C++語言

(3)仍保留了大量的C語言介面。

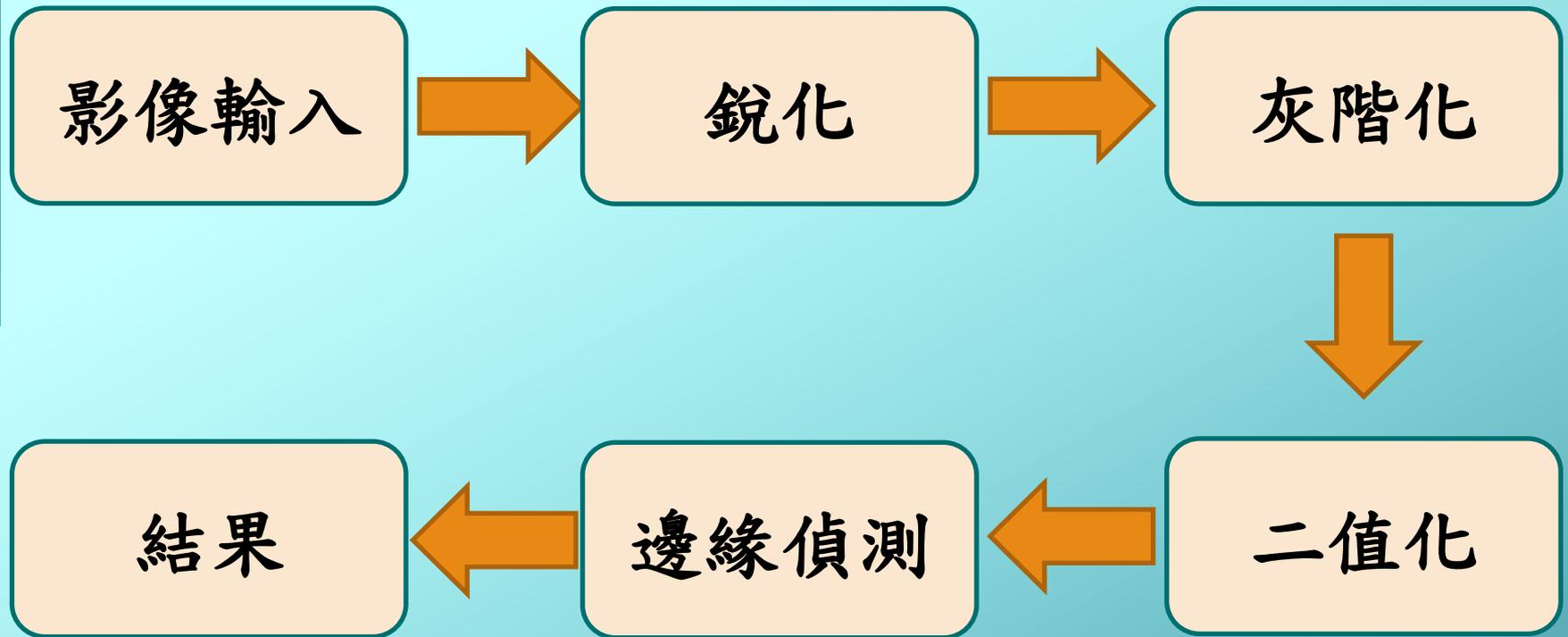
電腦視覺：

從圖像或者多維資料中取得「資訊」的人工智慧系統



三、原理介紹(2) OpenCV辨識

2. 影像辨識流程圖：

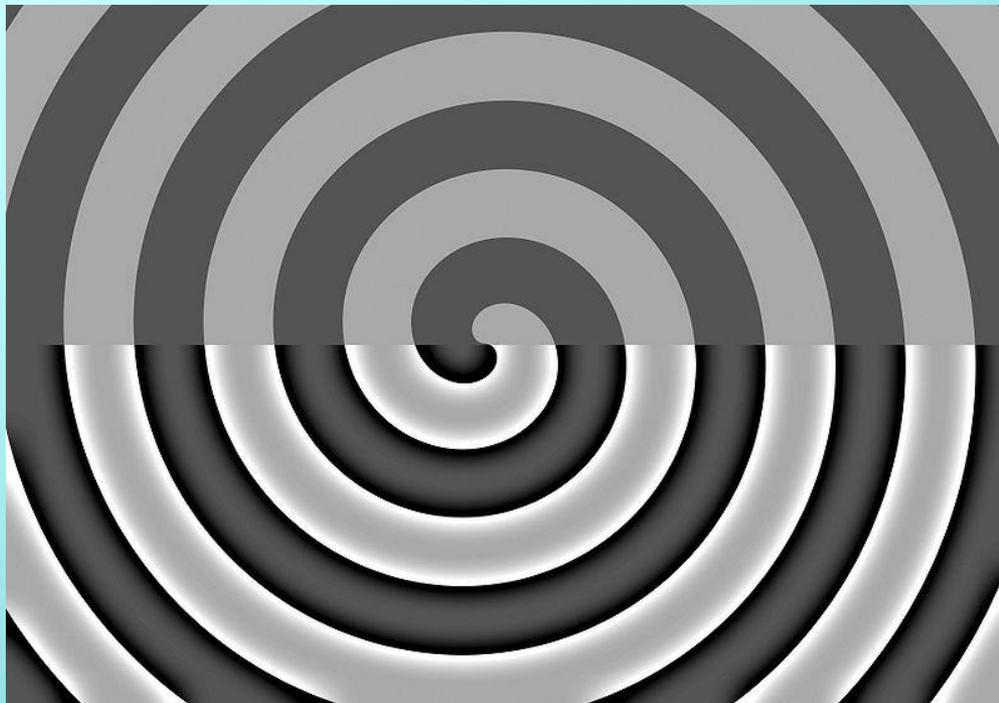


三、原理介紹(2) OpenCV辨識

3. 影像銳化：

(1)讓模糊的影像變得清晰，讓細節更明顯

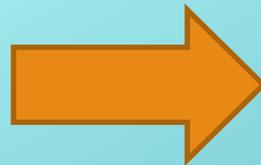
(2)銳化主要用於增強影像的強度變化



三、原理介紹(2) OpenCV辨識

4. 灰階化：

- (1) 每一個像素佔有的資料大小是一個位元組
- (2) 由最黑到最白之間可有**256**種明亮度
- (3) 有效簡化圖片資訊

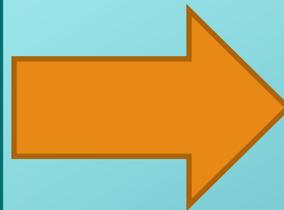


三、原理介紹(2) OpenCV辨識

5. 二值化：

- (1) 二值化主要的目的是降低彩色影像所帶來的大量資料量，而降低了處理的速度。
- (2) 可以將影像的前景與背景分開。

目的：區分出圖像中的物件與背景，以獲得物件的資訊



三、原理介紹(2) OpenCV辨識

6. canny邊緣偵測法

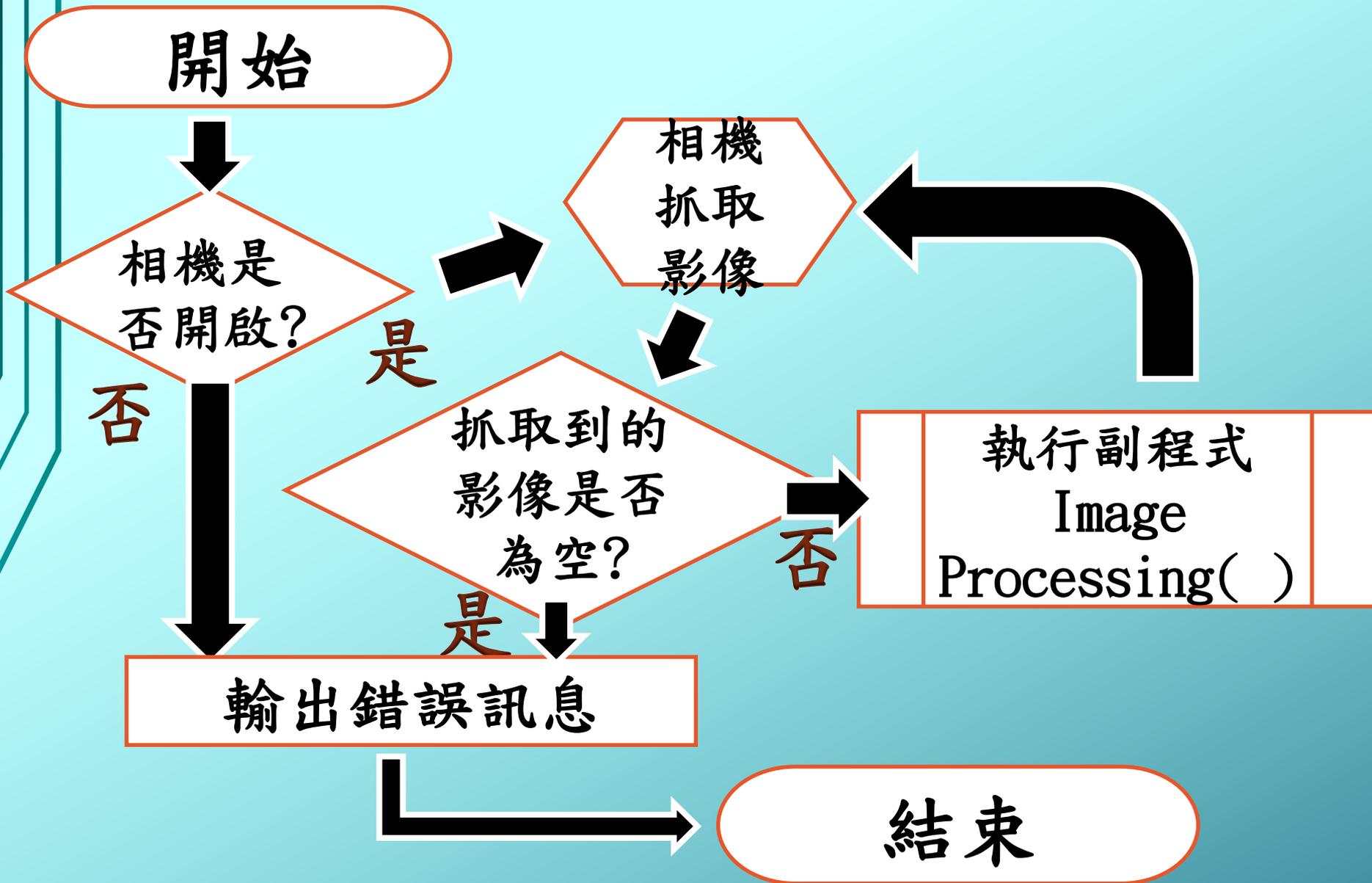
優點：

- (1)可以有效偵測圖片中所有邊緣
- (2)定位佳，以Canny的偵測誤差極小

缺點：

- (1)多重標度所偵測出的多重邊緣
- (2)無法有效分離出邊界輪框

五、程式介紹(1) 主程式



五、程式介紹(1) 主程式

(函式1)

```
#include "opencv2/core.hpp"
```

//儲存影像的資料結構、影像明亮、繪圖、
檔案處理、擷取攝影機相頭

```
#include "opencv2/objdetect.hpp" //基本物件偵測
```

```
#include "opencv2/videoio.hpp" //影像的輸出入
```

```
#include "opencv2/highgui.hpp"
```

//視窗介面或是鍵盤輸入操作功能

```
#include "opencv2/imgproc.hpp"
```

//影像的處理如：翻轉縮放、灰階、二值化、邊緣偵測等操作

五、程式介紹(1) 主程式

(函式2, 命名空間)

```
#include "opencv2/imgcodecs.hpp"
```

```
//影像的讀取和寫入
```

```
#include <iostream> //C++基本輸出入
```

```
#include <stdio.h> //C語言基本輸出入
```

```
#include <stdlib.h>; //C語言較為複雜的數學運算
```

```
using namespace std // C++ 的命名空間
```

```
using namespace cv; // OpenCV的命名空間
```

五、程式介紹(1) 主程式

(主程式1)

```
int main( void )
```

```
{
```

```
    VideoCapture capture; //宣告相機
```

```
    Mat frame;           //宣告影像
```

```
    capture.open( -1 ); //開啟相機
```

```
    //設定相機大小
```

```
    capture.set(CV_CAP_PROP_FRAME_WIDTH, Camera_W);
```

```
    capture.set(CV_CAP_PROP_FRAME_HEIGHT, Camera_H);
```

五、程式介紹(1) 主程式

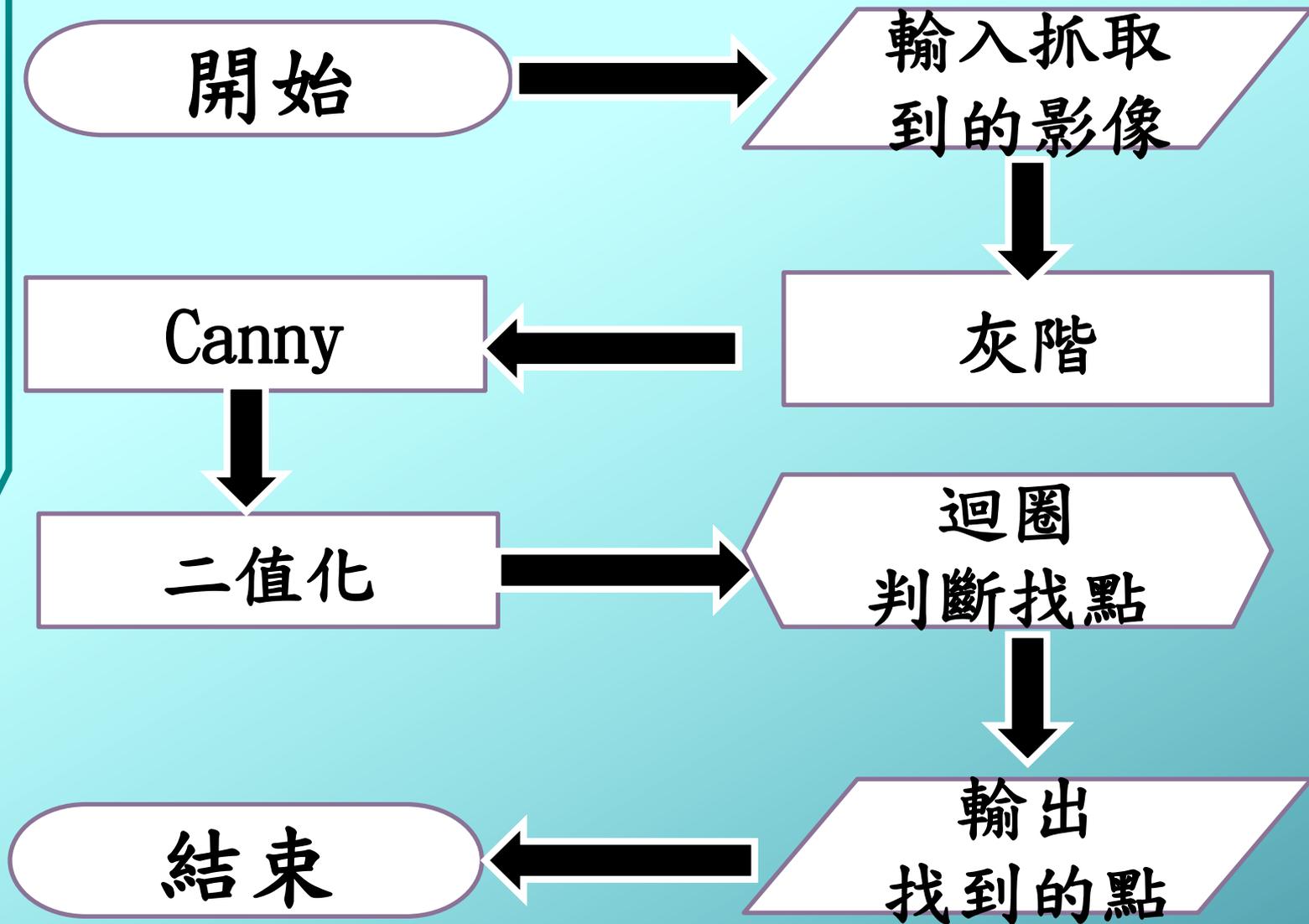
```
If ( ! capture.isOpened() ) //檢查相機是否開啟
{
    cout<<"*** Error opening Camera ***"<<endl;
    return -1; //跳出if並結束程式
}
```

五、程式介紹(1) 主程式

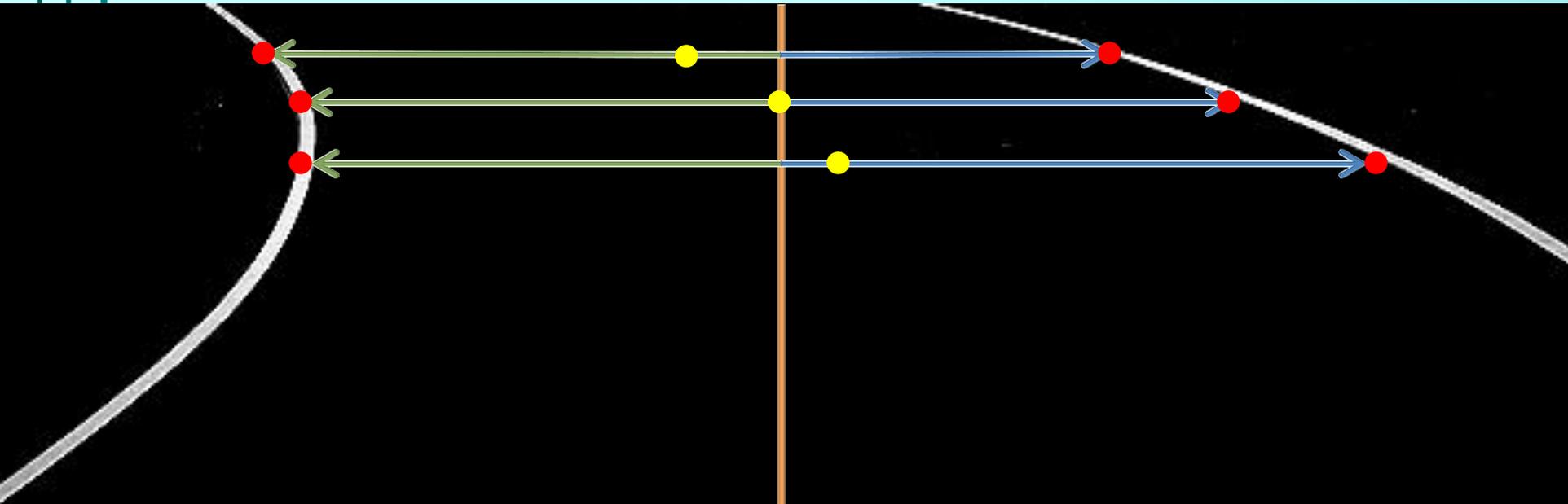
(主程式2)

```
while ( capture.read(frame) )
{
    if( frame.empty() )//檢查相機是否有接收到影像
    {
        cout<<"*** No Captured Frame ***" <<endl;
        break;//跳出while並結束程式
    }
    ImageProcessing( frame );//進入副程式    }
waitKey();
return 0;
}
```

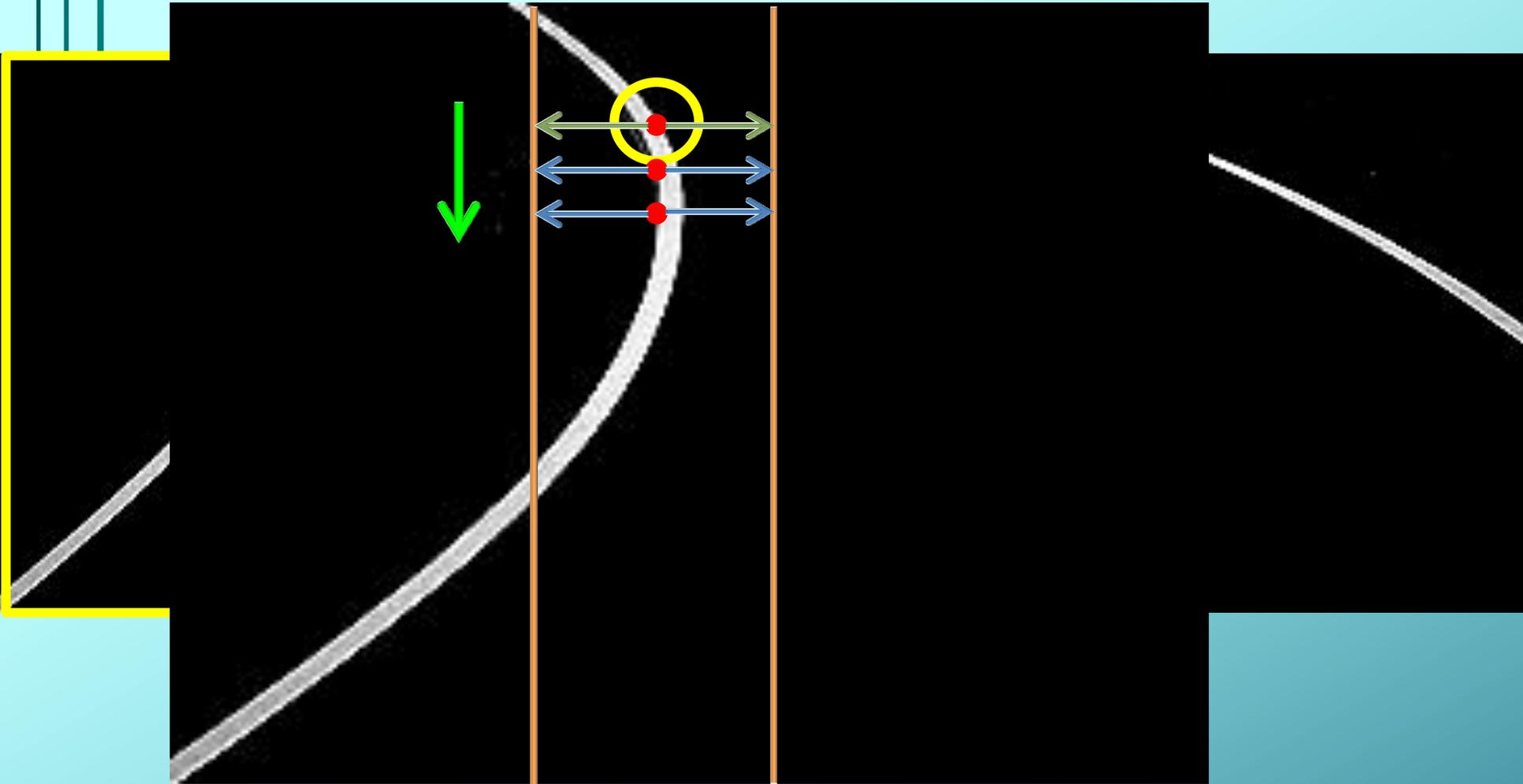
五、程式介紹(2) 副程式



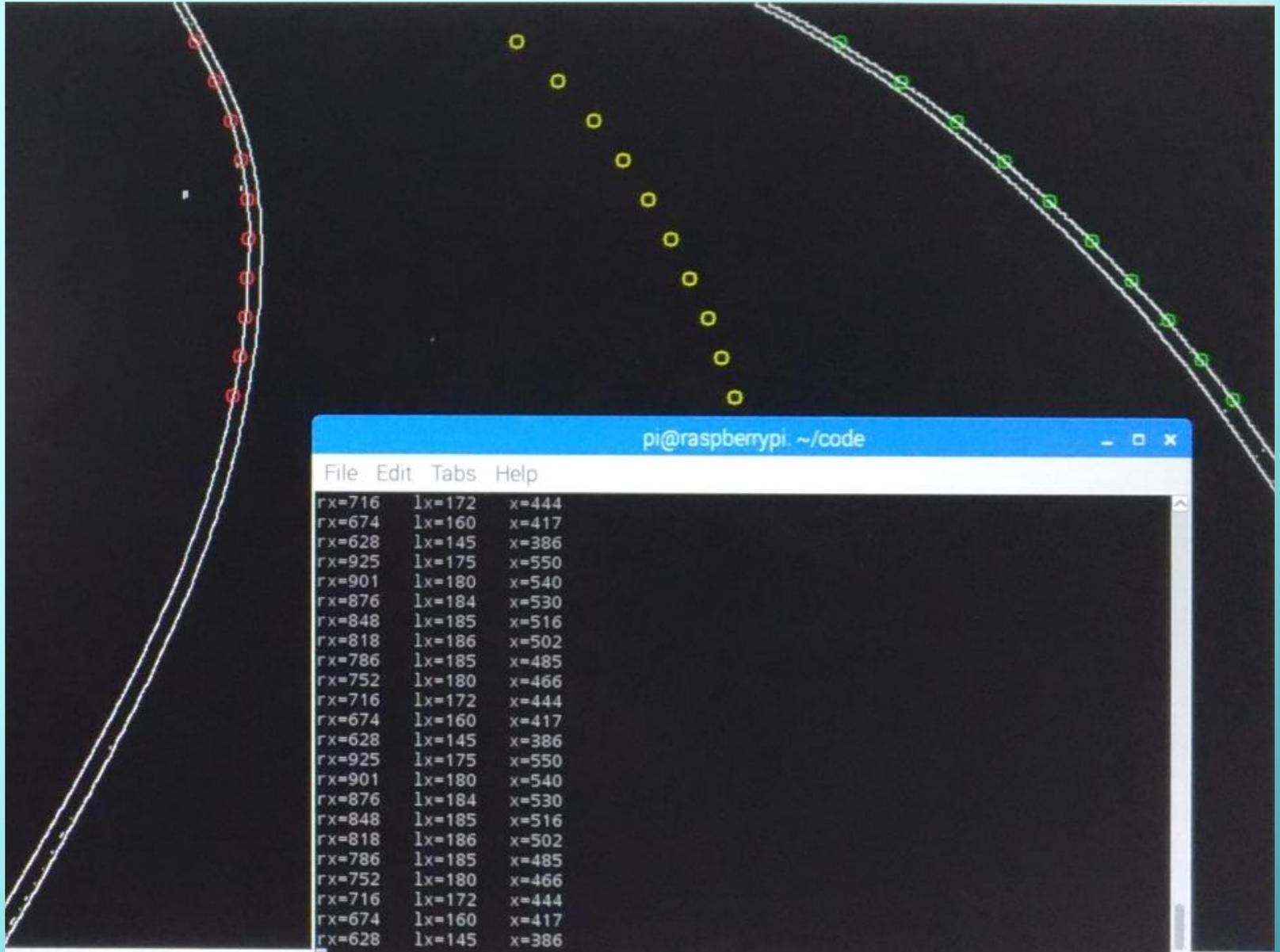
五、程式介紹(2) 副程式



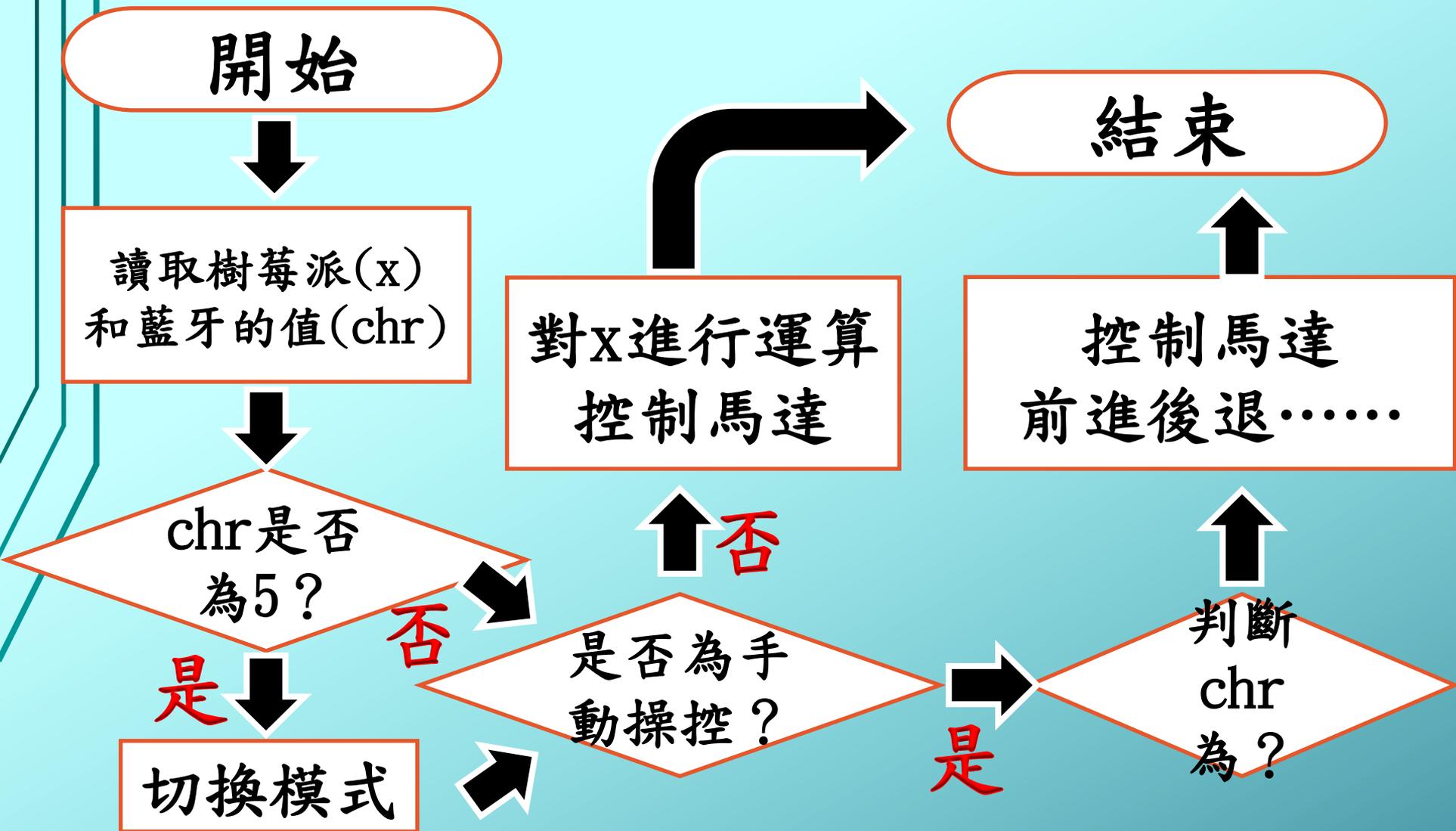
五、程式介紹(2) 副程式



五、程式介紹(3) 程式執行結果



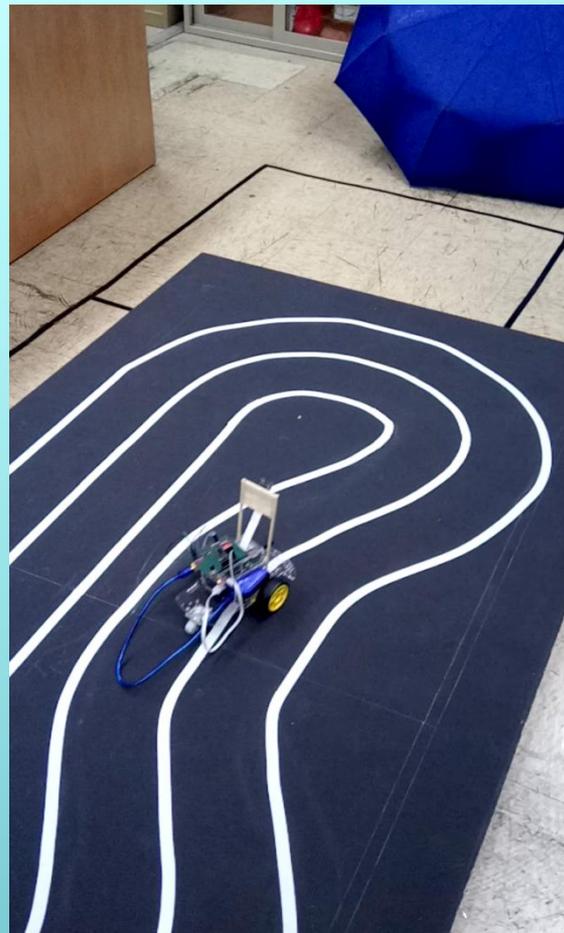
六、程式介紹-Arduino



七、成果影片



自動



遙控



八、結論與建議

我們學到了：

1. 影像辨識原理
2. 高斯濾波器
3. 樹莓派3的硬體配置
4. 樹莓派3程式撰寫(C++)

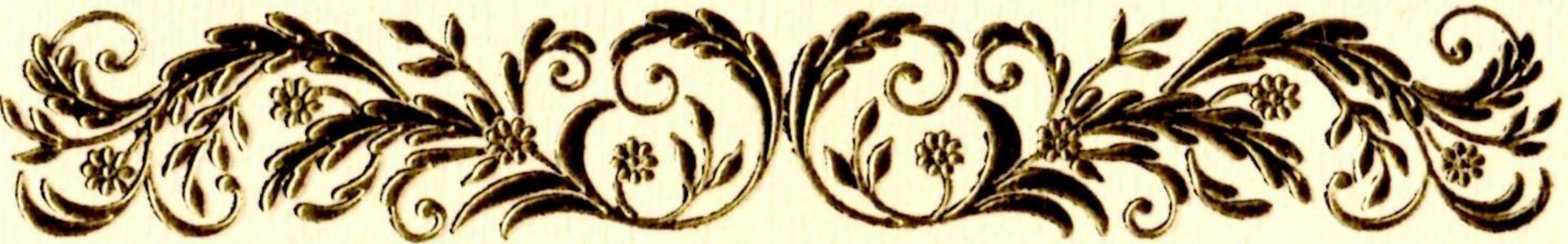
八、結論與建議

建議：

1. 量力而為
2. 利用廣角鏡頭
3. 前置作業study詳細一點
4. 善用藍牙修改參數，以提高做專題的效率

Q & A





Thank You

