

亞洲大學
資訊工程學系
105 學年度畢業專題

製具有即時控制及人臉辨識互動功能的樹梅派
機器人

指導老師：蔡志仁

組員：

張富循、黃品皓、徐偉恩、林宇智

專題功能

我們使用BB車、Raspberry pi 3、arduino以及python語言藉由simplecv函式庫裡的haar演算法，來作人臉辨識，抓取整個人臉面積及座標，傳送給arduino，由arduino隨著這些數值的變化，去達成人臉追蹤。

需求環境

- * Arduino，是一個開放原始碼的單晶片微控制器，採用開放原始碼的軟硬體平台，它可以控制很多種感測器另外它有USB介面，不需外接電源。另外有提供直流（DC）電源輸入。它的開發平台是arduino IDE，有LINUX系統版本與WINDOWS系統版本。
- * 樹莓派（Raspberry Pi），是一款基於Linux的單板機電腦，我們專題所使用的b行第三代，基本上它就像是一個電腦，因為我們與arduino互連所以我們選擇使用它。

需求環境

- * Python，是一種物件導向、直譯式的電腦程式語言。它包含了一組功能完備的標準庫，能夠輕鬆完成很多常見的任務。它的語法簡單，與其它大多數程式設計語言使用大括弧不一樣，它使用縮進來定義語句塊。
- * Python支援命令式程式設計、物件導向程式設計、函數式編程、面向側面的程式設計、泛型編程多種編程範式。
- * Arduino Software IDE
- * 使用與C語言和C++相仿的程式語言，並且提供了包含常見的輸入/輸出函式的Wiring軟體函式庫。一個典型的Arduino C/C++ sketch程式會包含兩個函式，它們會在編譯後合成為main()函式：
- * setup()：在程式執行開始時會執行一次，用於初始化設定。
- * loop()：直到Arduino硬體關閉前會重複執行函式內的程式碼。
- * 在使用GNU toolchain編譯和連結後，Arduino Software IDE提供了一個程式「avrdude」用來轉換可執行檔成為能夠燒寫入Arduino硬體的韌體。
- * 他能夠與很多設備連接，我們的樹梅派就是其中之一

需求環境

- * Simplecv 是一個開放式的函式庫，有了他，我們可以去處理高性能的影響處理，主要是在做影像辨識、圖片的格式轉換等等。
- * 哈爾特徵演算法（ Haar-like features ）
- * 是用於物體識別的一種數字圖像特徵。它們因為與哈爾小波轉換極為相似而得名，是第一種實時的人臉檢測算子，我們利用此演算法來時做我們的人臉辨識。

研究步驟及方法

- * 我們使用老師自己購買的bb車、攝影機、rasberry pi3、ardunio、迷你小主機、行動電源，組成一台樹梅派機器人，要來實作專題。
- * 第一步:先與老師溝通決定專題方向與用到設備列表
- * 第二步：練習基礎的影像辨識練習
- * 第三步：使用python套用simple作人臉辨識
- * 第四步：取人臉座標給arduino
- * 第五步：從arduino控制車子的移動，當人臉靠近車子往後退、人臉退後車子往前。

研究步驟及方法

- * 我們與老師溝通後，決定要做具有即時控制及人臉辨識互動功能的樹梅派機器人，首先，我們先在迷你小主機上練習matlab 影像辨識的基礎，之後因為老師覺得我們的進度太慢，讓我們先把主要功能做出，也就是人臉辨識互動的部分，我們所用的演算法是haar-feature，我們利用python裡可用的函式庫simple cv，裡面就能使用haar-feature，用此演算法抓取人臉座標，再將人臉分成x、y利用ser.write()傳送給arduino，由arduino 透過serial.read()接收從python傳來的座標及面積，讓車子隨著面積跟座標變化而動作。

程式碼

```
* Python
* from SimpleCV import Camera,Display,Image
* from time import sleep,time
* import serial
* ser = serial.Serial('/dev/ttyACM0',115200,timeout=1)#設定一個通道給arduino
* myCamera = Camera(prop_set={'width':160,'height':120})#設定顯示畫面的大小
* def facexya(frame):
*     faces = frame.findHaarFeatures('face.xml') #以人臉做辨識
*     if faces:
*         for face in faces:
*             print"Face at:"+str(face.coordinates()) #顯示出人臉座標
*             #w=face.width()
*             #h=face.height()
*             xmust = face.point[0][0]+face.x-(stache.width/2) #人臉座標x軸
*             ymust = face.point[0][1]+face.y+(stache.height/3) #人臉座標y軸
*             print "x:"+str(xmust)+" y:"+str(ymust)+" area:"+str(w*h) #印出x、y軸及面積
*             ixyj=str(w*h)+"_"+str(w)+"_"+str(h)+"_"+str(9)+"\n"#把取到的座標及面積透過str(9)傳給arduino
*             ser.write(ixyj)
*             ser.flush()
* while True:
*     try:
*         frame = myCamera.getImage()
*         facexya(frame)
*
```

程式碼

```
* arduino
*   if (word4=="9"){ // track face
*       x=word2.toInt(); 將x轉換為整數
*       area=word1.toInt();將area轉換為整數
*       if(area>4000) 當面積>4000 往後退
*           {backward(500);
*             }
*       if(area<-5000)
*           {
*           forward(500);
*           }
*       if(x>50)
*           turnRight(500);
*       }
*       else
*           turnLeft(500);
*       ball_axis_in="";
*   }
* }
```

研究成果

當我們的人臉靠近，車子會往後退；當人臉退後，車子會往前，向右偏車子就會向右偏，向左偏車子就會向左偏，會依照我們傳送給arduino的座標及面積來做對應的動作。

組員分工

張富循	林宇智	黃品皓	徐偉恩
python程式撰寫	Arduino程式撰寫	硬體組裝	報告製作