



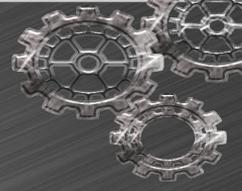
# 人工智慧在人臉辨識上的應用

成員：林紹綸 曾韋程 陳品丞 謝鼎璿

指導老師：楊仁元 老師

# 專題製作的背景及目的

我們選擇這個專題的原因是對於現代社會的科技蓬勃的發展，漸漸的科技已影響了我們的生活，並成了我們的一部分，科技的應用來自於人性，使我們的生活逐漸地方便化。基於科技便利化的理念，而選擇做此專題，我們希望能呈現給老師以及同學們看的功能包括：人臉辨識、機器學習此兩大功能。



程式語言皆使用Python

人臉偵測使用Opencv+dlib

機器學習

人臉偵測

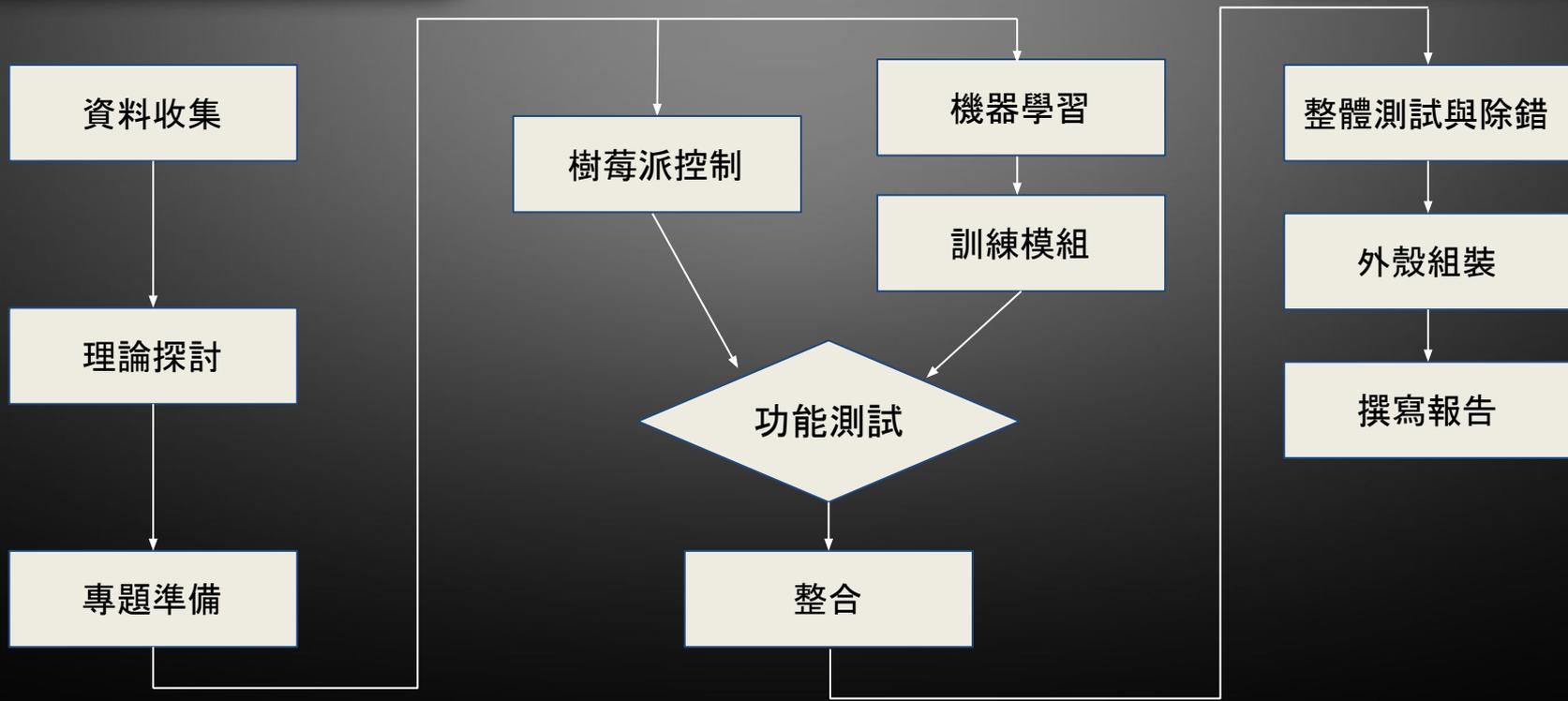
樹莓派控制

人臉辨識

七段顯示器+LED+按鈕

雷射切割外殼

# 製作流程圖



# 理論探討

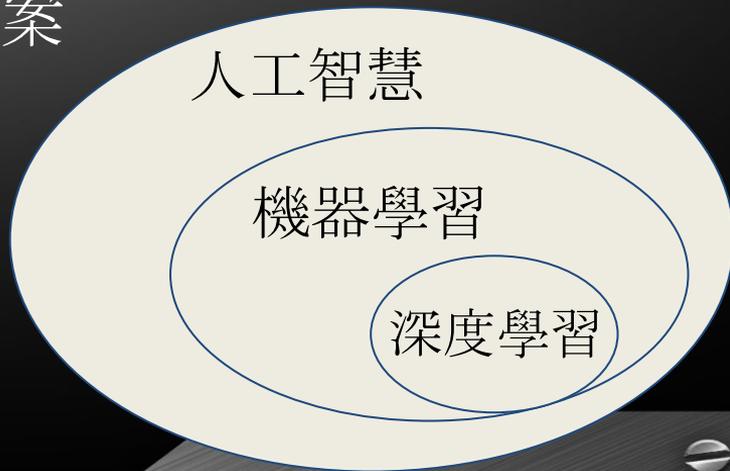
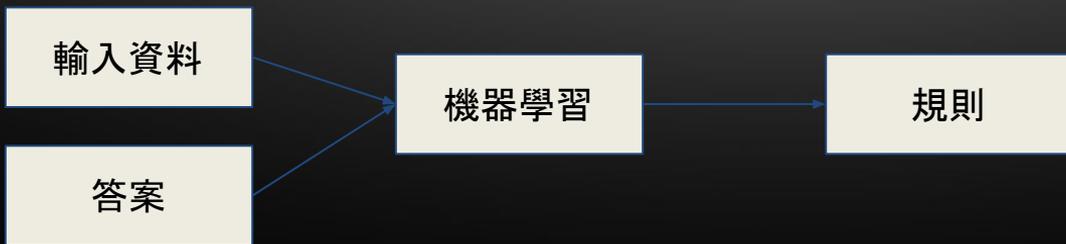
# 人工智慧 VS 機器學習



人工智慧的一般定義是：「能自動化地執行一般人類的智慧工作」。傳統的人工智慧開發只是手動輸入且定義大量的規則。



機器學習是被訓練的，是希望開發者不用手動建立規則，而是靠電腦自己從資料中尋找規則找出答案



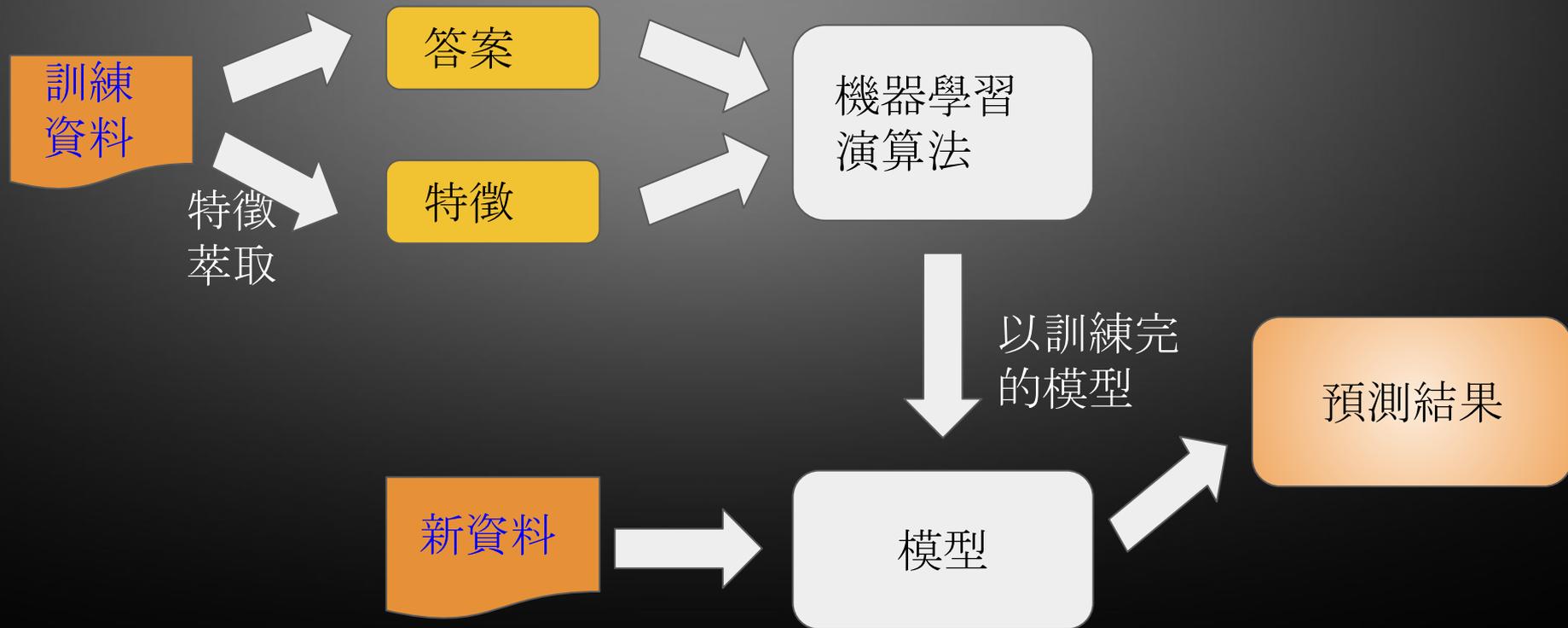
# Tensorflow Keras

TensorFlow是Google開發的開源機器學習工具。TensorFlow提供開發軟體庫，用於機器學習，用於研究和生產許多Google商業產品，如語音辨識、Gmail、Google相簿和搜尋。

Keras是一個專門用來訓練深度學習模型的API，能以TensorFlow、Theano為後端運行，導入不同模塊建立深度學習模型Keras的開發重點是支持快速的實驗。能夠以最小的時間把你的想法轉換為實驗結果，是做好研究的關鍵。



# 機器學習流程圖



# CNN卷積神經網路 VS RNN循環神經網路

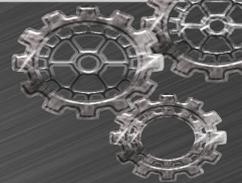
CNN是由一個輸入層一個或多個隱藏層和一個輸出層所組成，CNN以數字來表現像素，按照這種方式來分辨多種特徵以獲得答案

RNN可以用於一系列資料以猜測接下來會發生的內容，而過去可能有使用類似句子來訓練RNN的記憶功能，使得RNN可以預測接下來的文字內容

簡單來說，**CNN適用於影像辨識**，**RNN用於文字預測自然語言和語音辨識**

# CNN 卷積神經網路

## (Convolutional Neural Network)

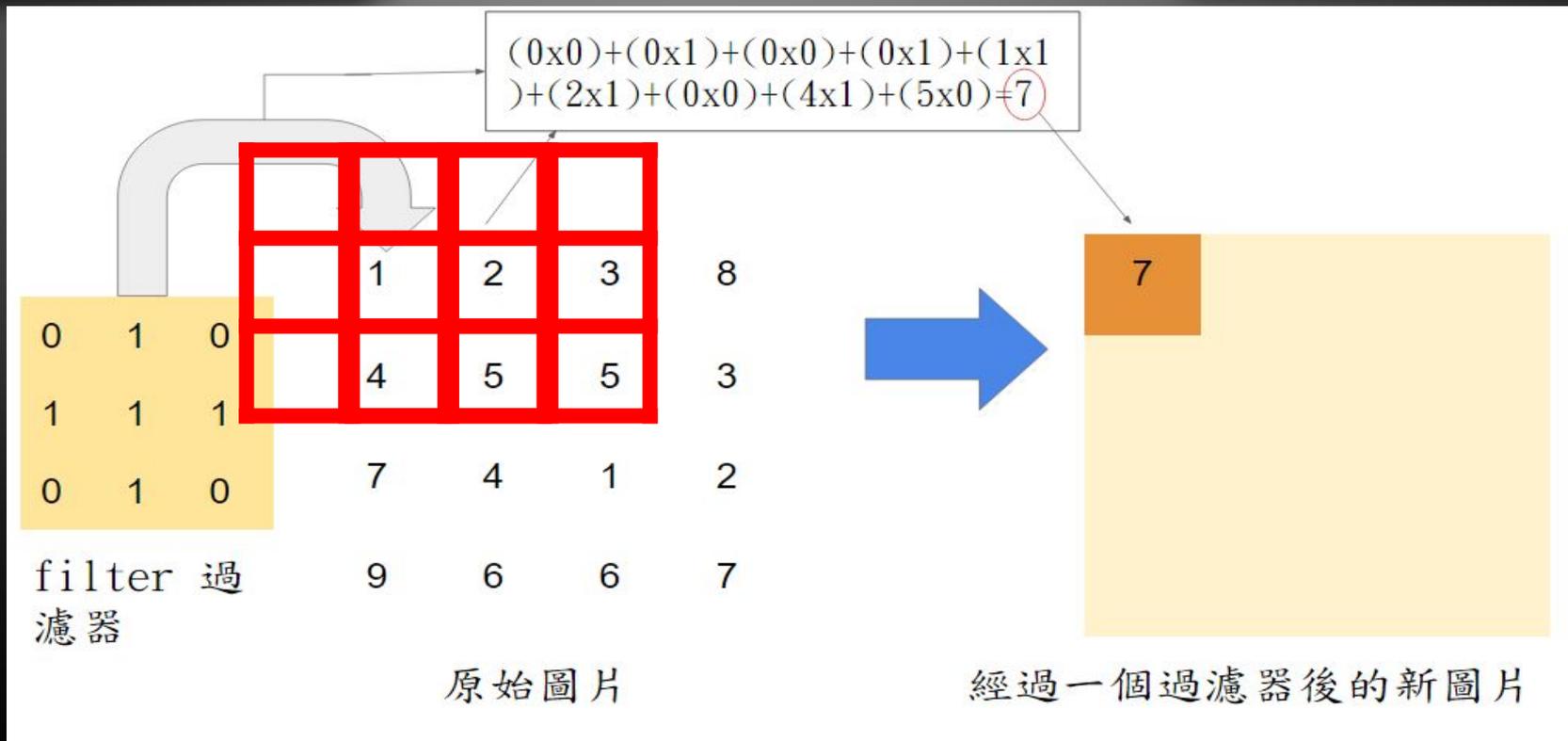


與MLP(多層感知器)的主要差異是:CNN增加了卷積層、池化層的去提取特徵。

卷積層的意義是將原本一個影像經過卷積計算,產生多個不同重要特徵的影像。

池化層是將經過卷積的樣本作縮減取樣,比如將 $28*28$ 的影像轉為 $14*14$ 的影像,但原本的樣本數量不變。這樣可以有效地提取重要特徵,捨棄不需要的部分,在後面的運算上提升速度與準確度。

# 卷積運算方式





# 池化運算方式

4	4	2	2
2	5	1	4
1	2	3	5
8	6	7	9

4個數字最大是5, 所以  
計算結果是5

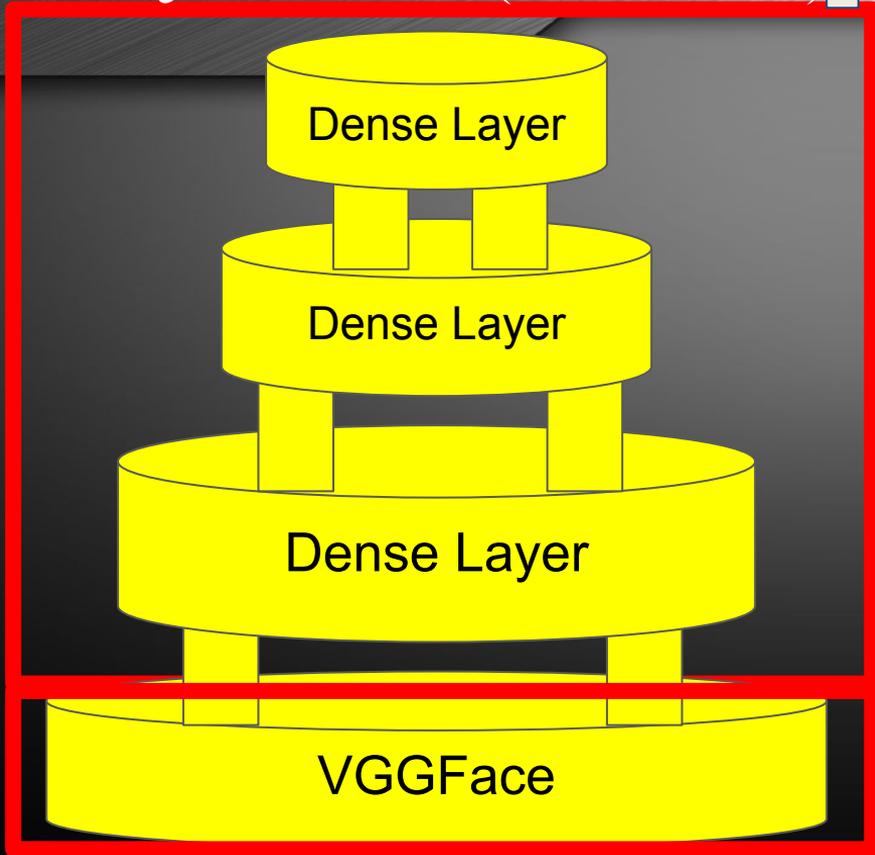
5	4
8	9

4個數字最大是9,  
所以計算結果是9

5	4
8	9

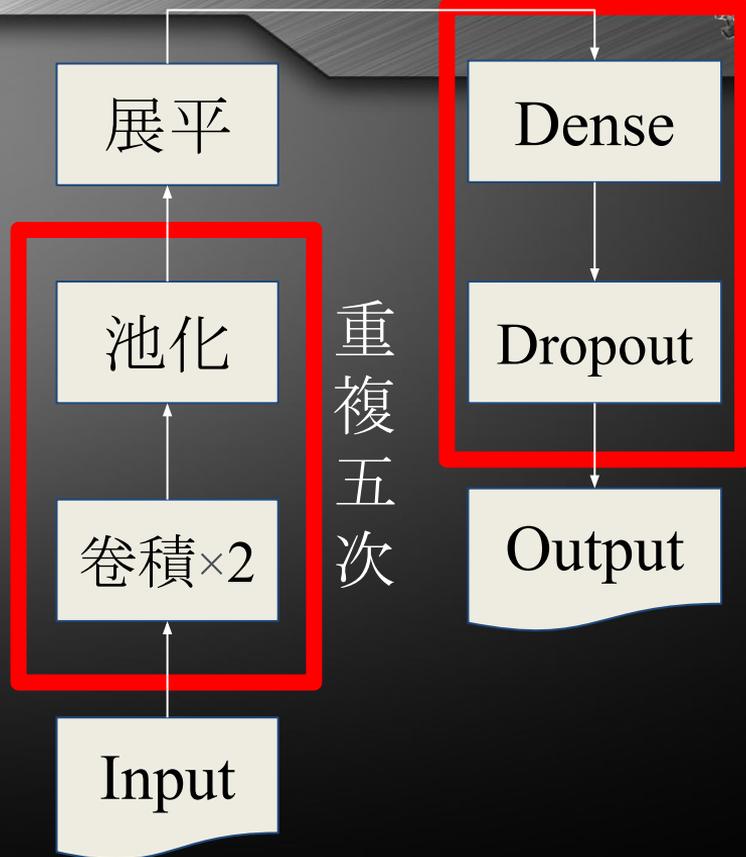
# Layer

MLP(多層感知器)



CNN卷積網

LOGO

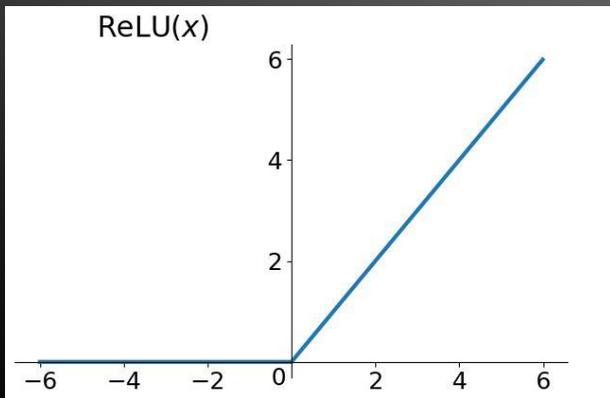


重複三次

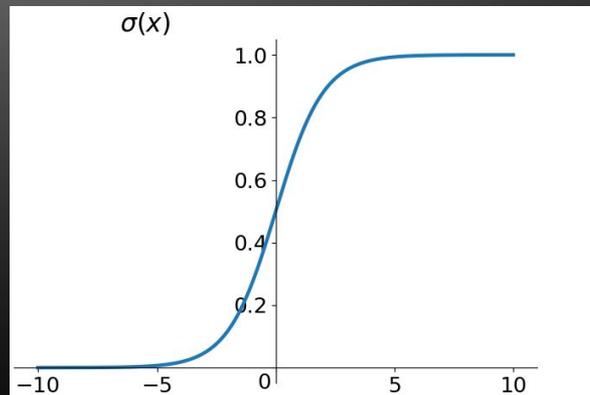
重複五次

# 激活函數(active function)

在類神經網路中如果不使用激活函數，那麼在類神經網路中皆是以上層輸入的線性組合作為這一層的輸出，這樣上層的輸出與這層的輸出依然脫離不了線性關係，所以必須經過激活函數才能脫離線性關係，我們這次所用的激活函數有用到兩種：

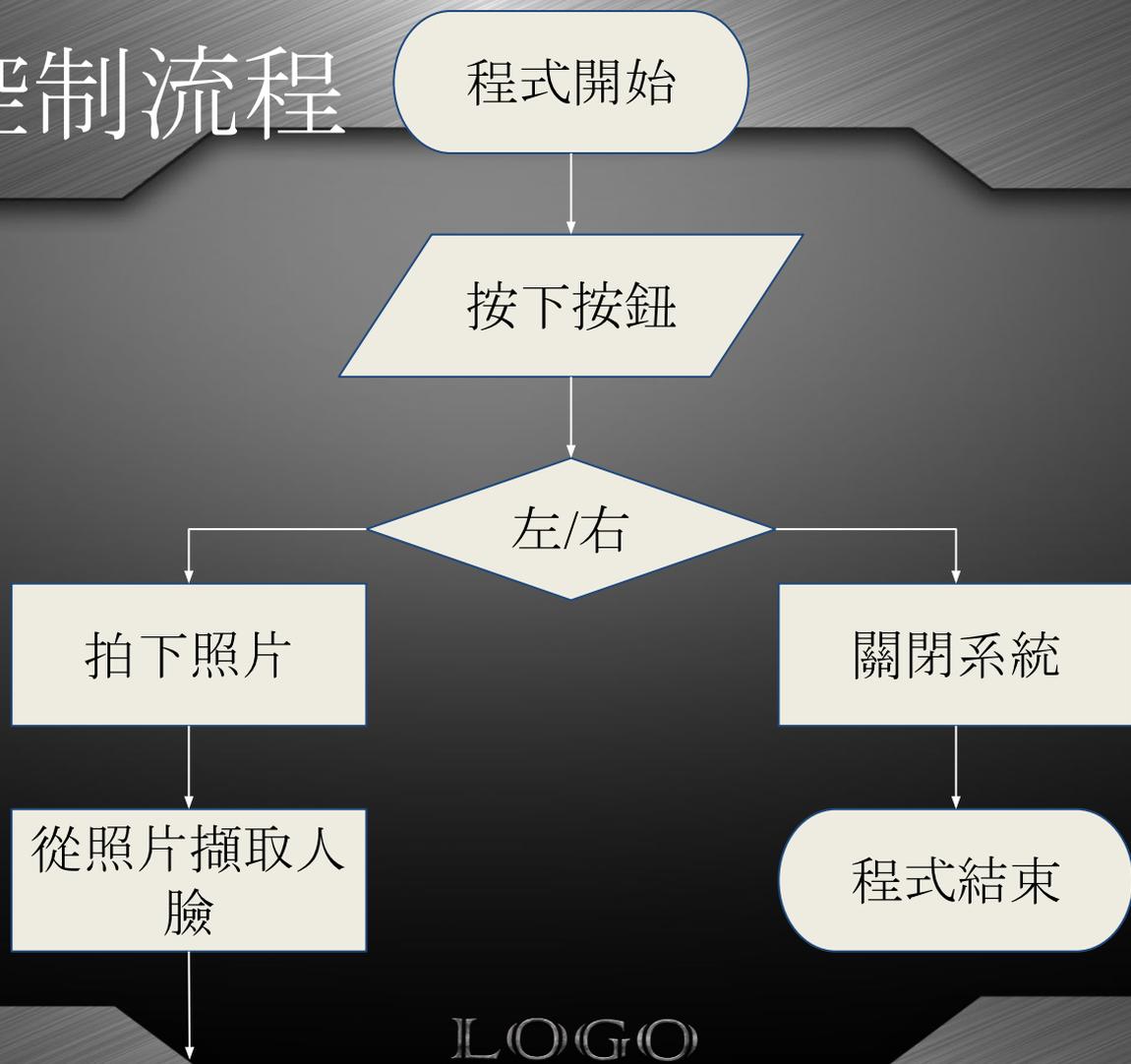


$$\text{ReLU}(x)=[0, \infty]$$



$$\text{sigmoid}(x)=[0, 1]$$

# 樹莓派控制流程

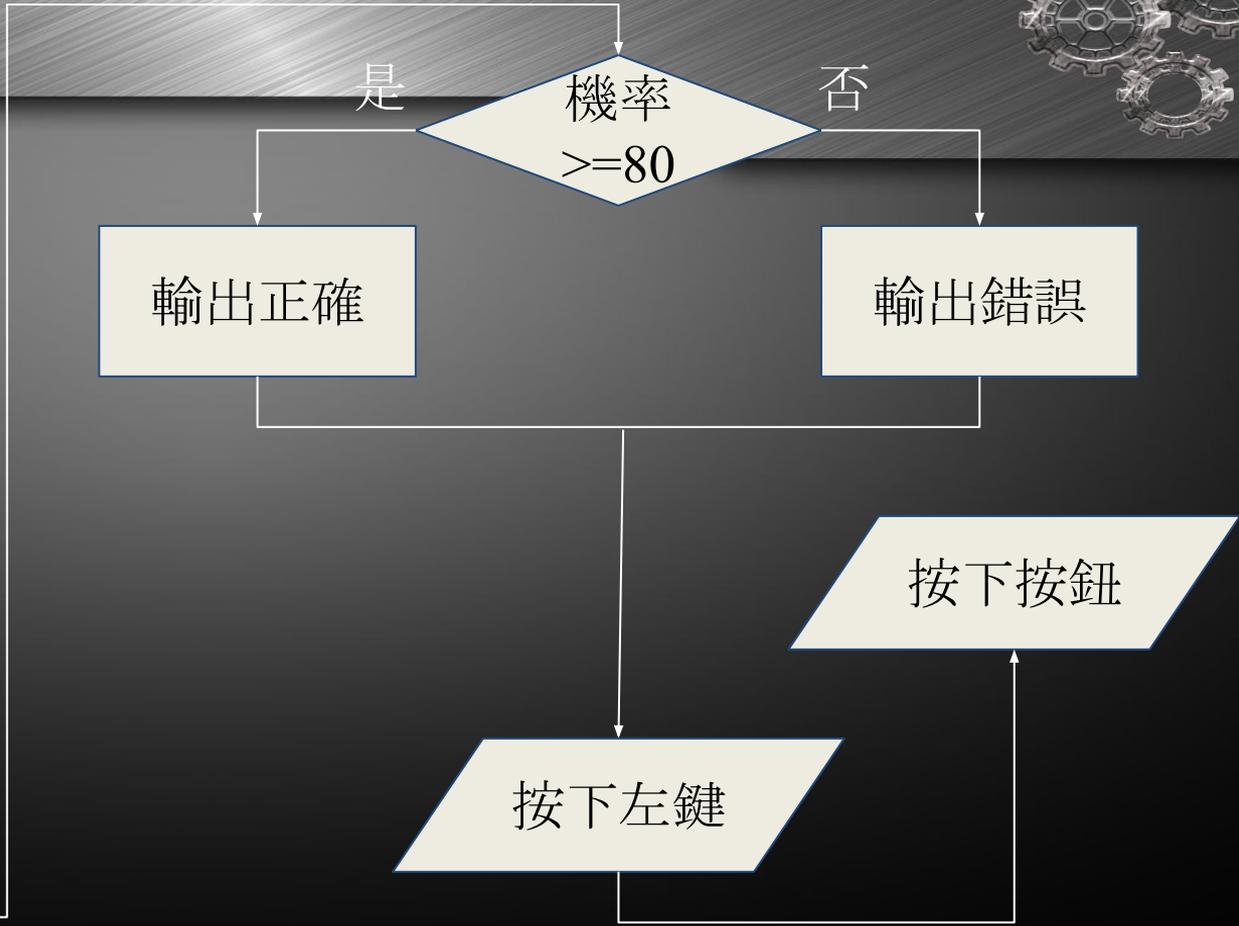




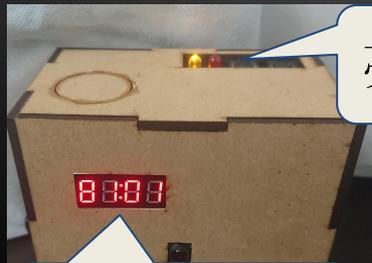
輸出所有  
預測值

選出最大值與  
其標籤

印出機率與標  
籤

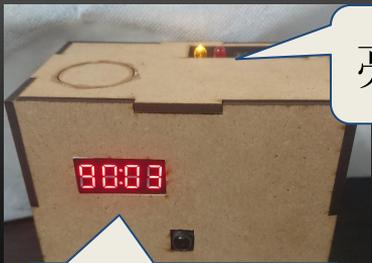


# 功能展示



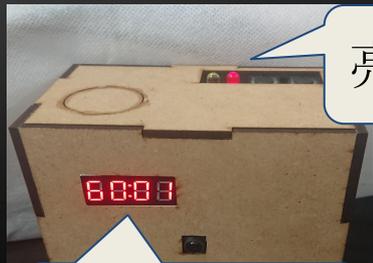
亮黃燈

81%/標籤01  
正確



亮黃燈

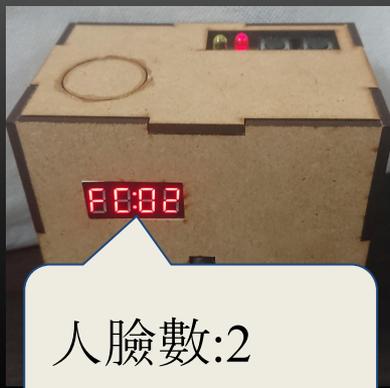
90%/標籤03  
正確



亮紅燈

60%/標籤01  
錯誤

藉由dlib的人臉偵測功能所做的過濾



# 遇到的困難與解決方法

1.從圖片中擷取人臉失敗，擁有人臉的圖片卻沒辦法從中找到人臉，或是找到的部分不是人臉。

A:原先只使用Opencv的人臉分類器，但因準確率太低改用dlib去做偵測，效果改善許多。

2.預測出的值不夠準確或是預測錯誤。

A:擴大CNN讓它提取的特徵更精細，或者使用其他人已訓練過的卷積層(VGGFace);給予更多資料去訓練;使用其他學習法,優化器,損失函數,調整卷積內部設定等等。



3.在樹莓派上安裝所需的系統時與Windows系統上的安裝方式天差地遠。

A:從網路上搜尋資料尋找正確的安裝方式。

4.機器學習的理論方面難以理解。

A:因為使用的是Keras, 在設定與訓練過程中都是使用官方配套好的訓練方法, 只要從中選擇適合自己要用的演算法和訓練器等等並做些微調整就能開始訓練了。

5.蒐集人臉的資料不容易

A:因為機器學習需要輸入大量的資料, 而因成本、時間、設備等有限, 所以我們的人臉都是用拍影片且藉由Opencv去擷取上6000張的照片, 雖每一幀的照片有些許不同但整體幾乎一樣, 這也是造成預測機率低的原因之一



Q&A



LOGO

# 參考資料

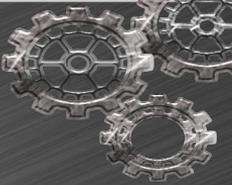
Keras中文文檔:<https://keras.io/zh/>

以深度學習進行人臉辨識:[http://www.cc.ntu.edu.tw/chinese/epaper/0043/20171220\\_4306.html](http://www.cc.ntu.edu.tw/chinese/epaper/0043/20171220_4306.html)

TensorFlow+Keras 深度學習 人工智慧實務應用 作者:林大貴

Deep learning 深度學習必讀:Keras 大神帶你用 Python 實作  
作者:François Chollet 譯者:葉欣睿

CNN入門講解:什麼是激活函數:  
<https://zhuanlan.zhihu.com/p/32824193>



【AI入門】深度學習概要與Keras實作:

[https://makerpro.cc/events/machine\\_learning2019/](https://makerpro.cc/events/machine_learning2019/)

CNN 與 RNN 之間的差

異? :<https://blogs.nvidia.com.tw/2018/09/05/whats-the-difference-between-a-cnn-and-an-rnn/>

基於python語言使用OpenCV搭配dlib實作人臉偵測與辨

識:<https://www.tpisoftware.com/tpu/articleDetails/950>

**Day 05: Keras 模型、函數及參數使用說明:**<https://ithelp.ithome.com.tw/articles/10191725>

謝謝各位的聆聽

LOGO