

# 課堂活動4: CNN網路其他應用

曾宏鈞 06160485

徐友笙 05360365





蔡毓丞 06370136

盧君彥 05360153

建立小組員辨識系統:

- 收集小組員的照片
- 利用VGGFace模型, 建立向量庫
- 測試辨認照片是否為某某小組員或者都不是

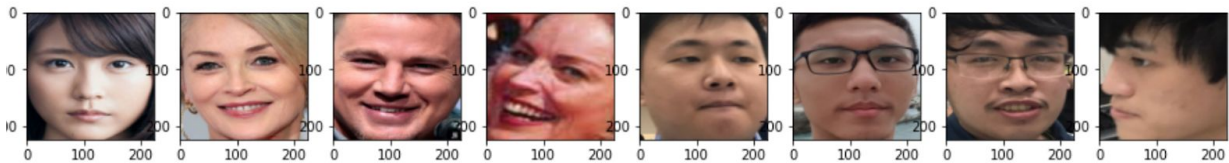
## 1. 小組員照片

A	C
	
D	B
	

## 2. 建立向量庫

```
[[0.      1.1111474  0.      ... 2.5270243  1.913619  0.      ]
 [0.11362483 2.1536467  0.      ... 2.7116737  1.5266435  0.11776159]
 [0.12404148 2.2267408  0.      ... 1.4977454  0.36678237  0.73817205]
 ...
 [0.431913  0.      0.      ... 1.2728522  0.      0.01679866]
 [0.01755741 0.6569582  1.2290447 ... 0.1902724  0.02267258  0.      ]
 [0.      0.0139272  0.02281416 ... 8.253765  0.      3.0090048 ]]
shape : (50, 2048)
```

## 3. 測試辨認照片是否為某某小組員或者都不是



```
===== test/toshiba-dynabook.jpg =====
japanese_actress
===== test/sharon_stone3.jpg =====
not in the dataset
===== test/channing_tatum.jpg =====
not in the dataset
===== test/sharon_stone2.jpg =====
not in the dataset
===== test/Data_200524_0025.jpg =====
A
===== test/Data_200524_0037.jpg =====
D
===== test/Data_200524_0003.jpg =====
C
===== test/Data_200524_0013.jpg =====
B
```

## 實驗步驟：

使用目前state-of-art的臉部視覺演算法 Multi-Task Cascaded Convolutional Neural Network(mtcnn) 擷取臉部,

## [Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks.](#)

### Step1 下載

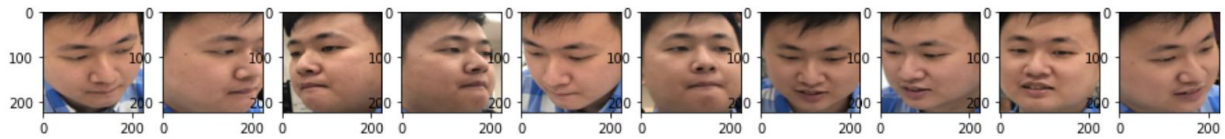
```
sudo pip install mtcnn
```

```
pip install git+https://github.com/rcmalli/keras-vggface.git
```

...

### Step2 使用mtcnn擷取小組員臉部特徵

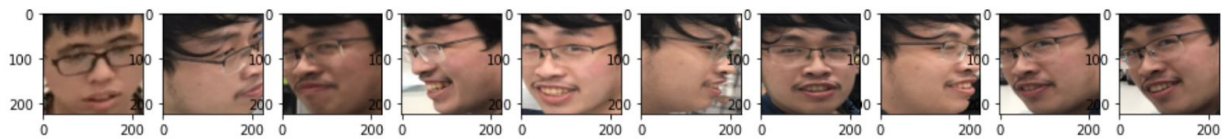
class: A



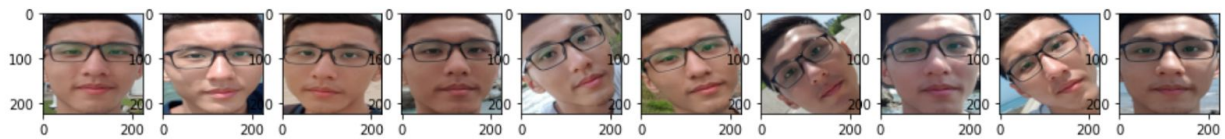
class: B



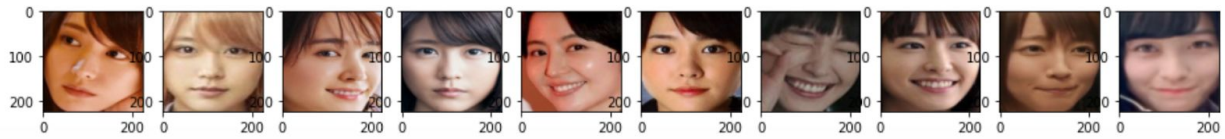
class: C



class: D



class: japanese\_actress



### Step3 face embedding

```
[[0.      1.1111474  0.      ... 2.5270243  1.913619  0.      ]
 [0.11362483 2.1536467 0.      ... 2.7116737  1.5266435  0.11776159]
 [0.12404148 2.2267408 0.      ... 1.4977454  0.36678237  0.73817205]
 ...
 [0.431913  0.      0.      ... 1.2728522  0.      0.01679866]
 [0.01755741 0.6569582 1.2290447 ... 0.1902724  0.02267258  0.      ]
 [0.      0.0139272  0.02281416 ... 8.253765  0.      3.0090048 ]]
shape : (50, 2048)
```

把每一個類別的embedding vector，去算n(2048)維平均向量，方便之後計算向量夾角。

### Step4 看cos(θ)(向量夾角)

因為向量夾角愈小，代表兩向量愈接近，因此我們使用1-cos(θ)，去做score，因為cos值域是[-1,1]，我們將門檻值設定為0.4，大於0.4才去判斷他是哪一個類別(跟每一個類別取score取最大值)。

設在一個2048維的特徵向量中，已知5種類別(A、B、C、D、japanese\_actress)他們的平均向量為：

$$C_1 : m_1 = [x_{1,1}, x_{1,2}, x_{1,3} \dots x_{1,2048}]$$

$$C_2 : m_2 = [x_{2,1}, x_{2,2}, x_{2,3} \dots x_{2,2048}]$$

$$C_3 : m_3 = [x_{3,1}, x_{3,2}, x_{3,3} \dots x_{3,2048}]$$

...

若對於輸入的樣本  $X_0 = [x_{0,1}, x_{0,2} \dots x_{0,2048}]$  而言，現以兩向量的夾角大小作為判斷，則得

$$\frac{X_0^T \cdot m_i}{\|X_0\| \|m_i\|} = \cos \theta \equiv a_i, \quad i = 1, 2, 3, 4, 5$$

在此

$$\|X_0\| = \sqrt{x_{0,1}^2 + x_{0,2}^2 \dots x_{0,2048}^2}$$

$$\|m_i\| = \sqrt{x_{i,1}^2 + x_{i,2}^2 \dots x_{i,2048}^2}$$

判斷法則：

$$X_0 \in C_i, \text{ 當 } j = \underset{i}{\operatorname{argmin}} \{a_i\}, \quad i = 1, 2, 3, 4, 5$$

否則  $X_0 \notin \{C_1, C_2, C_3, C_4, C_5\}$

# 筆記

Face recognition(人臉辨認) 有兩種主要步驟

1. Face Verification (人臉驗證)  
1對1, 將input跟已知的身份比對 ex. 是不是人臉
2. Face Identification (人臉識別)  
1對多, 與人臉資料庫比對 ex. 這個人是誰

VGGFace Model [\[Deep Face Recognition 2015\]](#)

特點：

- 使用深層CNN
- 使用很大的CNN測試集(Google Image Search)

模型步驟

## Step1 (learning)

先使用softmax為output的activation function

## Step 2 (face embedding)

拿掉softmax層, 直接輸出代表臉部的特徵向量

## Step 3 (fine-tuning)

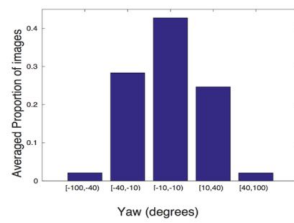
微調模型訓練, 使得相同的人歐式距離越小, 不同的歐式距離愈大

VGGFace2

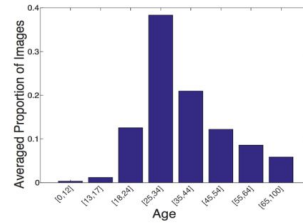
[\[VGGFace2: A dataset for recognizing faces across pose and age 2017\]](#)

特點

- 331萬張9131個類別(每類別平均362張)
- 每類別差異很大(姿勢、年齡、曝光度、種族、職業)



(a) pose statistics



(b) age statistics



(c) John Wesley Shipp



(d) Leymah Gbowee



(e) Princess Haya Bint Al Hussein



(f) Julio César Chávez Jr.



(g) Roy Jones Jr.



(h) Ruby Lin



(i) Additi Gupta



(j) Lee Joon-gi

- 使用resnet-50&SqueezeNet效果超過之前的state-of-art模型

## 模型步驟

### Step 1 (face embedding 臉部嵌入)

使用模型預測給定的臉部特徵向量（參考VGGface）為2048個不同長度的向量

### Step 2 (face descriptor 臉部描述)

使用 L2 vector norm (與原本點的歐式距離) 將向量正規化為1或單位向量長度,

### Step 3 (計算餘弦)

與step2的face descriptor算餘弦

註：face descriptor是一個經L2正規化過、2048維的descriptor，是由分類層提取的

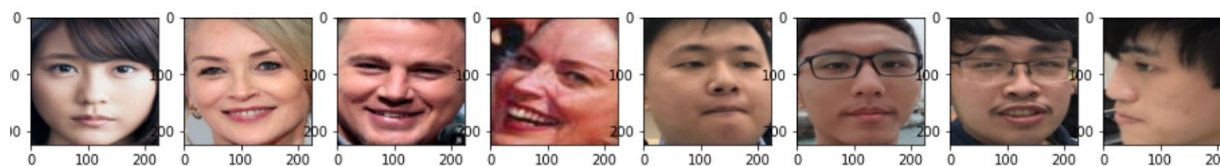
## 參考

[How to Perform Face Recognition With VGGFace2 in Keras](#)



## 心得

這次的作業我們閱讀了臉部辨識的兩個state-of-art的深度學習模型 - VGGface、VGGface2，可看出兩者皆是先去使用mtcnn去做臉部擷取後，找到人臉後，再將做**embedding**成特徵向量，藉由特徵類別的高維向量到測試圖片特徵的餘弦距離去辨別是否之前有在資料庫裡面，這樣就可以利用別人很好的model去做臉部的驗證，也較不會有分類錯誤的問題。根據實驗結果顯示，也有成功分類出小組員。雖然觀察mtcnn的結果後發現若一張圖片有多張人的時候容易抓到別人的臉，因此在準備資料的時候應該要找只有一個人的，閱讀完這篇論文也比較知道別人怎麼做**fine-tuning**的。也可知道其實VGG-face做了一些multi-tasking的技巧，原來是可以讓模型有更好的效能！



```
===== test/toshiba-dynabook.jpg =====
japanese_actress
===== test/sharon_stone3.jpg =====
not in the dataset
===== test/channing_tatum.jpg =====
not in the dataset
===== test/sharon_stone2.jpg =====
not in the dataset
===== test/Data_200524_0025.jpg =====
A
===== test/Data_200524_0037.jpg =====
D
===== test/Data_200524_0003.jpg =====
C
===== test/Data_200524_0013.jpg =====
B
```