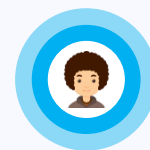
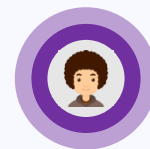


IIE FINAL PROJECT

預測人臉年齡



110034558 廖文辰



授課教授：邱銘傳 博士

目錄

01

研究背景

02

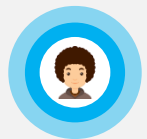
研究方法

03

模型訓練

04

結論與未來展望

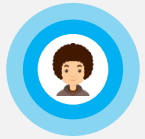


研究背景

世界各地有形形色色的人種，每個人也有著自己獨特的樣貌，人隨著年齡增長外貌也會有著些許變化。在這個網路無遠弗屆的時代，人們往往會經營自己的社群網站，甚至是遊玩交友軟體，因此人們都會放上自己的大頭貼，讓別人間接認識自己。

本研究將以CNN model建立人臉年齡識別系統，幫助使用者藉由人臉照片辨識人臉年齡。除了娛樂一般民眾，對於未來此相關發展也有著重大意義。





研究背景

5W1H

Who：一般民眾、學術單位

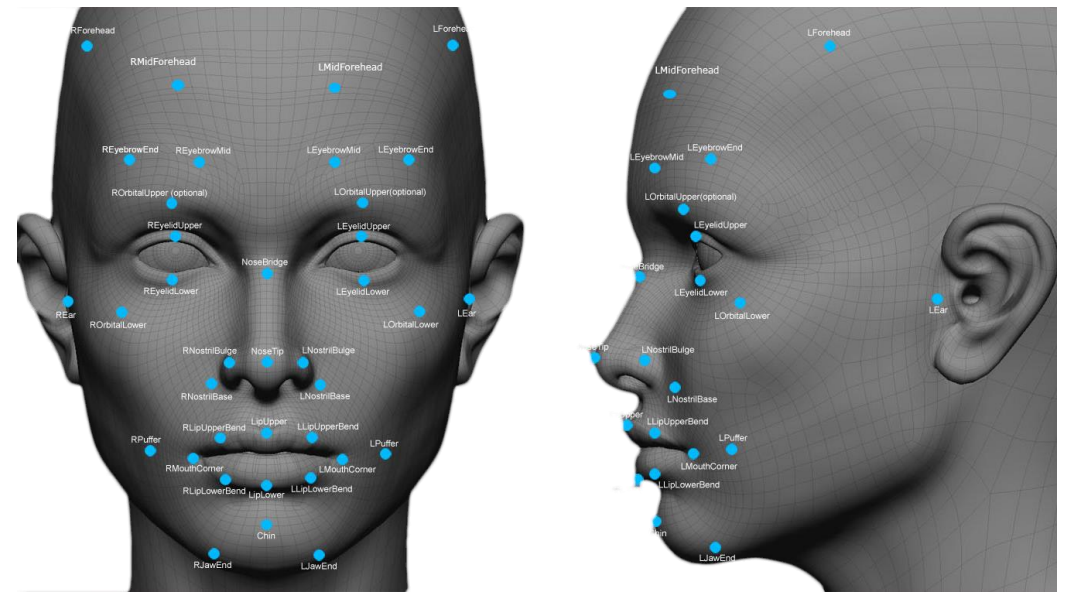
What：更精準預估人臉年齡

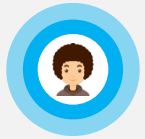
Why：提高現實中人機互動系統的應用需求

When：看到照片時

Where：世界各地

How：以CNN深度學習建立識別模型





研究方法-資料來源

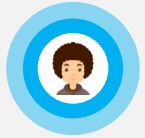
本資料集使用 Kaggle 之 Data Every Day Dataset ,
資料集被劃分為訓練集與測試集。

資料集中包含了20至50歲各年齡人像圖片。

訓練集33462張訓練用圖片。

測試集7008張影像。





研究方法-資料前處理

插入numpy, pandas...等常見第三方套件與模組，
並且從雲端匯入訓練集中的33462張訓練用圖片
及測試集中的7008張影像。

```
import numpy as np
import pandas as pd
from pathlib import Path
import os.path

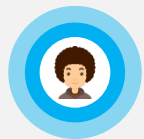
from sklearn.model_selection import train_test_split #用sklearn將train和test資料分開

import tensorflow as tf #用tensorflow建立圖片資料產生器和模型

from sklearn.metrics import r2_score #用R平方分數衡量模型的好壞
```

```
from google.colab import drive
drive.mount('/content/drive')

image_dir = Path('/content/drive/MyDrive/20-50/')
from google.colab import files
```



研究方法-模型建立

- 建立FilePath和Age
- 把兩者合併為「image」

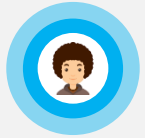


▶ images

↳

	FilePath	Age
0	/content/drive/MyDrive/20-50/train/22/113386.jpg	22
1	/content/drive/MyDrive/20-50/train/29/170287.jpg	29
2	/content/drive/MyDrive/20-50/train/41/179421.jpg	41
3	/content/drive/MyDrive/20-50/train/47/126188.jpg	47
4	/content/drive/MyDrive/20-50/train/30/167003.jpg	30
...
40435	/content/drive/MyDrive/20-50/train/27/174935.jpg	27
40436	/content/drive/MyDrive/20-50/train/49/144643.jpg	49
40437	/content/drive/MyDrive/20-50/train/25/147638.jpg	25
40438	/content/drive/MyDrive/20-50/train/33/146050.jpg	33
40439	/content/drive/MyDrive/20-50/train/49/142203.jpg	49

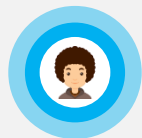
40440 rows x 2 columns



研究方法-模型建立

- 以7:3的比例將資料分割為訓練資料與測試資料。
- 隨機抽取數據並用random_state可重複展現相同的隨機結果。

```
image_df = images.sample(5000, random_state=1).reset_index(drop=True)
train_df, test_df = train_test_split(image_df, train_size=0.7, shuffle=True, random_state=1)
```

研究方法-載入圖片

train_df

	Filepath	Age
1334	/content/drive/MyDrive/20-50/train/26/162593.jpg	26
4768	/content/drive/MyDrive/20-50/train/28/147733.jpg	28
65	/content/drive/MyDrive/20-50/test/45/42885.jpg	45
177	/content/drive/MyDrive/20-50/train/27/178438.jpg	27
4489	/content/drive/MyDrive/20-50/test/31/44435.jpg	31
...
2895	/content/drive/MyDrive/20-50/train/25/166220.jpg	25
2763	/content/drive/MyDrive/20-50/train/23/174142.jpg	23
905	/content/drive/MyDrive/20-50/test/48/43433.jpg	48
3980	/content/drive/MyDrive/20-50/test/33/42502.jpg	33
235	/content/drive/MyDrive/20-50/train/22/114475.jpg	22

3500 rows × 2 columns

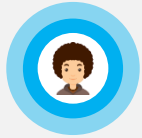


```
train_generator = tf.keras.preprocessing.image.ImageDataGenerator(
    rescale=1./255,
    validation_split=0.2
)

test_generator = tf.keras.preprocessing.image.ImageDataGenerator(
    rescale=1./255
)
```

```
train_images = train_generator.flow_from_dataframe(
    dataframe=train_df,
    x_col='Filepath',
    y_col='Age',
    target_size=(120, 120), #像素縮到120*120 加快訓練時間
    color_mode='rgb',
    class_mode='raw',
    batch_size=32,
    shuffle=True,
    seed=42,
    subset='training'
)
```

```
Found 2800 validated image filenames.
Found 700 validated image filenames.
Found 1500 validated image filenames.
```



模型訓練與訓練結果-模型訓練

第一層卷積層filter為16，kernel size為3*3
第二層卷積層filter為32，kernel size為3*3
activation皆設為relu。

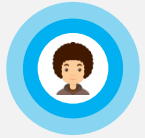
Optimizer : adam
Loss : mse
Epochs : 100

```
inputs = tf.keras.Input(shape=(120, 120, 3))
x = tf.keras.layers.Conv2D(filters=16, kernel_size=(3, 3), activation='relu')(inputs)
x = tf.keras.layers.MaxPool2D()(x)
x = tf.keras.layers.Conv2D(filters=32, kernel_size=(3, 3), activation='relu')(x)
x = tf.keras.layers.MaxPool2D()(x)
x = tf.keras.layers.GlobalAveragePooling2D()(x)
x = tf.keras.layers.Dense(64, activation='relu')(x)
x = tf.keras.layers.Dense(64, activation='relu')(x)
outputs = tf.keras.layers.Dense(1, activation='linear')(x)

model = tf.keras.Model(inputs=inputs, outputs=outputs)

model.compile(
    optimizer='adam',
    loss='mse'
)

history = model.fit(
    train_images,
    validation_data=val_images,
    epochs=100,
    callbacks=[
        tf.keras.callbacks.EarlyStopping(
            monitor='val_loss',
            patience=5,
            restore_best_weights=True
        )
    ]
)
```



模型訓練與訓練結果-訓練結果與參數優化

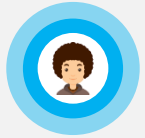
```
predicted_ages = np.squeeze(model.predict(test_images))
true_ages = test_images.labels

rmse = np.sqrt(model.evaluate(test_images, verbose=0))
print("    Test RMSE: {:.5f}".format(rmse))

r2 = r2_score(true_ages, predicted_ages)
print("Test R^2 Score: {:.5f}".format(r2))
```

Test RMSE: 8.82989
Test R² Score: -0.00056

epoch	100	50	10
RMSE	8.82989	8.82743	8.84744



結論與未來展望

結論

訓練結果沒有想像中的好之可能原因

- 卷積層太少
- 資料集不夠多

未來展望

若臉部識別的技術更加成熟，未來也許能對性取向、犯罪傾向與暴力傾向等議題之預測能有更進一步的發展。





Thank you!!

