中華民國第 65 屆中小學科學展覽會

作品說明書

高級中等學校組　電腦與資訊學科

052510

全「面」分析：基於面部表情的情緒模型

學校名稱：國立彰化高級中學

作者：	指導老師：
高二 周宥勝 高二 黃振哲	邱科文

關鍵詞：深度學習、臉部表情、情緒狀態

摘要

本研究著重於透過面部表情的判斷進而推斷情緒狀態的深度學習神經網路模型。首先，將人類的情緒狀態分為七個類別，並透過 MediaPipe 框架取得影像中的臉部特徵位置，利用其座標框定臉部範圍並作為深度學習模型的輸入。第二，透過部不同特徵訓練模型，進一步優化模型識別情緒的準確度，並嘗試不同的特徵，例如：趨勢曲線函數的係數或是各種向量。最後我們設計了一個插件，利用此模型在 Google Meet 中進行即時的影像辨識並分析，作為線上課程的授課成效評估。

目前本模型在透過五層卷積層配合池化層及全連階層，並提取瞳孔眉毛向量、瞳孔鼻尖向量以及口部中心為基準點的向量作為特徵，能夠有 84%的準確度，F1 值落在約 85，其中快樂情緒的辨識精準度高達 98%。

壹、 前言

一、研究動機

在人類日常生活中，情緒對於行為與決策扮演著關鍵角色，影響著我們的學習效率、購物選擇、社交互動，甚至是對娛樂活動的體驗。然而，情緒的變化往往是潛在的，難以單純透過語言或主觀回饋準確衡量。因此，如何客觀地分析人們在特定活動中的情緒反應，成為一個值得探討的問題。

人類的面部表情會因不同的情緒狀態而產生微妙變化，例如眉毛的抬升代表驚訝，嘴角的上揚表示快樂，皺眉則可能與焦慮或不滿相關，隨著科技的發展，我們可以透過影像數據分析這些特徵變化，可以獲得一群人在特定情境下的情緒分佈，進一步評估特定活動帶來的情感影響。

本研究希望應用臉部情緒辨識技術，分析人們在特定活動中的情緒變化，並作為評估活動體驗的工具。例如：在前陣子的線上教學期間，老師無法像實體教學時能夠時時刻刻掌握所有同學的狀況，因此我們想利用深度學習模型協助老師進行線上課程的成效評估。同樣的技術亦可以應用於其他的領域，像是在電影放映期間，觀眾的面部表情可反映他們對劇情的情緒反應，進而分析該電影是否成功帶來預期的情感效果。若能進一步統計不同觀眾群體在特定片段的情緒變化，甚至可作為電影製作或行銷決策的參考。

二、研究目的

（一）基於 Mediapipe 框架，在不同光照、角度和遮蔽情況下，準確偵測並標記影像中人臉的 468 個特徵點。

（二）利用深度學習神經網路框架訓練出的情緒辨識模型準確率達 80%以上。

（三）面部表情辨識的模型可以準確的辨識出使用者的七種面部表情，分別為憤怒（angry）、厭惡（disgust）、恐懼（fear）、快樂（happy）、悲傷（sad）、驚訝（surprise）和中性（neutral）。

（四）建立一個基於卷積神經網路（CNN）的面部表情辨識模型，用於評估使用者在線上課程中的情緒狀態。

三、文獻回顧

（一）面部表情與情緒狀態的關係

面部表情與情緒狀態有著直接的關係，除了明顯的正負向的表情（快樂與生氣）外，臉部的一些細微表情也會顯現一個人的情緒狀態，而情緒狀態表達的就是自身內心的想法、接受或者厭惡[7][8]。本研究透過面部表情的觀察推估情緒狀態，進而瞭解受試者的內心感受，進一步做分析。

（二）深度學習（Deep Learning）

深度學習是機器學習中的一個關鍵分支，其基於模擬人類大腦神經網絡結構來進行數據的分析與處理。這一技術的核心是人工神經網絡，其中由多層神經元組成，每一層的神經元透過激活函數和權重連接。隨著層數的增加，這些網絡能夠學習並提取數據中更為複雜的特徵，從而提升數據處理和預測的能力。深度學習技術已在語音識別、圖像處理等多個領域取得顯著成果。在文獻[2]中有提及各種深度學習的方式對面部表情領域的成效。而本研究會選擇深度學習作為訓練模型的方式。

（三）卷積神經網路（Convolutional Neural Network，即 CNN）

卷積神經網絡是深度學習中的一種重要架構，專門用於處理具有網格結構的數據。其主要特點是通過卷積層進行特徵提取，利用權重共享和池化操作來減少計算量並提高學習效率。卷積層的核心是 filter，一個小的矩陣，會在圖像上滑動並進行卷積運算，提取圖像中的局部特徵，如邊緣、角落等。每個 filter 會學習到不同的特徵，並在訓練過程中自動調整，以捕捉圖像中的不同模式。隨著層數的增加，CNN 能夠學習到越來越複雜的特徵，從而提升數據處理和預測的能力。此外，CNN 還包含全連接層，用於最終的分類決策。而本研究將參考文獻[1]中提及的方式，改良並訓練模型。

（四）激活函數（Activation function）

激活函數在神經網絡中負責將每個神經元的輸入轉換為輸出，並引入非線性特徵，這樣神經網絡才能學習到複雜的數據模式。透過激活函數，網絡能夠擁有足夠的能力來處理非線性問題，提升模型的表現。它對神經網絡的學習效率、精度和收斂速度有著重要影響。選擇合適的激活函數能夠避免訓練過程中的困難，如梯度問題，並幫助模型有效解決實際應用中的問題。在文獻[2]中有提及不同激活函數可以滿足的需求，本研究將會與文獻[1]使用相同的 ReLU 函數作為激活函數。

$$ReLU(x) = max(x, 0)$$

（五）梯度下降法（Gradient Descent）

梯度下降法是一種常用的優化技術，目的是透過不斷調整模型參數來最小化損失函數，從而提升模型的準確度。其核心思想是計算損失函數對模型參數的梯度，並沿著梯度的反方向進行調整。每次更新參數時，都會根據梯度的大小和方向來縮小損失函數的值。梯度下降法有幾個版本，分別是批量梯度下降、隨機梯度下降和小批量梯度下降，它們主要區別在於每次參數更新時所使用的訓練數據量。這些變體可以幫助加速訓練過程，提高計算效率[1][2][5]。本研究也將使用此方法作為模型訓練的優化方式。

（六）面目特徵的提取

主動外觀模型(AAM)、Haar 類似特徵(Haar)、局部二元模式(LBP)[2]、方向梯度直方圖(HOG)的技術[1]、像素值分析 (Pixel Analysis)[5]、深度信念網路 (Deep Belief Networks, DBN)[2]、遞迴神經網路 (Recurrent Neural Networks, RNN)[5]、生成對抗網路 (Generative Adversarial Networks, GAN)[2]、多模態方法 [2]、注意力機制 (Attention Mechanism)[2]，而本研究是使用 Mediapipe 標定臉部地標（Facial Landmarks）去做特徵提取。

（七）面部表情的分類

由 Paul Ekman 提出的面部表情分類，在多數論文中皆被提及[1][2][4][6]。面部表情可以分為七種，分別為憤怒（angry）、厭惡（disgust）、恐懼（fear）、快樂（happy）、悲傷（sad）、驚訝（surprise）和中性（neutral）。本研究也是以此七種表情做為分類。

並且各種情緒有在論文中有分別提及可能的面部表情特徵。本研究會基於這七種情緒作為分類的標準，下表為情緒可能會伴隨的面部表情特徵之整理：

情緒	可能的面部特徵
快樂	嘴角可能上揚，眼睛會張的較開
憤怒	眉毛較靠近眼睛、眼睛較小
厭惡	無提及
難過	眉毛下垂，嘴巴為水平或凹向下
恐懼	左右眉毛較靠近、嘴巴可能張開
中性	無提及
驚訝	嘴巴可能張大，眼睛張的較開

表一、情緒可能伴隨的面部表情特徵（圖表來源：作者自行製作）

（八）混淆矩陣（Confusion Matrix）

混淆矩陣可以作為一種模型評估的數據，其運作原理是將每種預測值與實際值做成對照的表格，並分成四個部分，分別為正確預測(True)/錯誤預測(False)對應正確圖片(Positive)/錯誤圖片(Negative)，產生 TP、FP、TN、FN 四個值，並可以由此去計算出更多的評估指標，包含 Accuracy（A）、Precision（P）、Recall（R）與 F1 值，其中 F1 最常被用來作為做模型評估的指標，其值為 P 和 R 的調和平均數，在文獻[1][6]中有提及，本研究將使用混淆矩陣作為模型的成效指標。

$\text{Accuracy(A)} = \frac{TP + TN}{TP + TN + FP + FN}$	$Precision(P) = \frac{TP}{TP + FP}$
$\text{Recall}(R) = \frac{TP}{TP + FN}$	$F_1 = 2(\frac{PR}{P + R})$

表二、各種評估指標的計算方式（圖表來源：作者自行製作）

參、研究設備及器材

一、硬體設備與環境

（一）筆記型電腦

1. 作業系統：Windows 11

2. CPU：Intel Core i7-12700H

3. GPU：GeForce RTX 3050Ti

（二）Anaconda

Anaconda 內含大量常用的數據分析與科學計算模組，如 NumPy、Pandas 和 TensorFlow。內建套件管理工具，方便用戶管理環境與安裝函式庫，能夠簡單的避免不同套件的版本污染。

二、軟體設備

（一）Python

Python 是一種語法簡單的高階程式語言。其中包含許多方便的函式庫，在本研究中主要處理數據處理分析與模型訓練。

1. Keras：支援多種深度學習框架，如 TensorFlow，並且內建多種常用的優化器和損失函數，適合用於快速實驗和原型設計。

2. Mediapipe：提供高效的機器學習框架，包含多種預訓練的模型、面部識別、手部追蹤等功能。

3. Numpy：一個強大的數值運算庫，主要用於處理大型、多維的數據，並提供高效率的數學運算和矩陣操作功能。

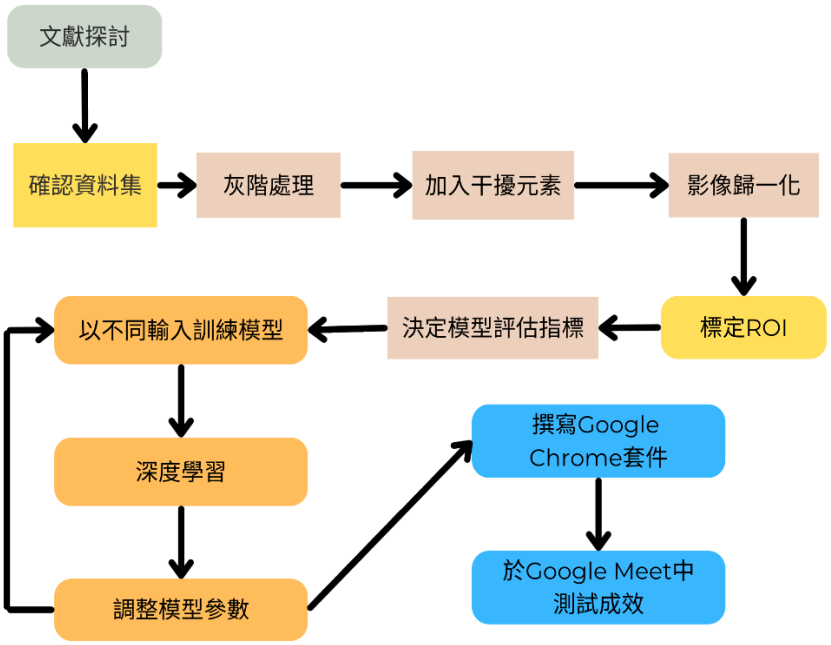
4. matplotlib：是一個強大的資料視覺化庫，用於創建靜態、動態和交互式的圖表。

三、Facial Expression Traing Data 資料集（FETD）

Kaggle 上的 FETD 資料集包含了 27331 張 96×96 的彩色面部表情圖像，包含了開心（Happy）4524 張、驚訝（Surprise）3997 張、難過（Sad）3020 張、中立（Neutral）4508 張、憤怒（Anger）3121 張、害怕（Fear）3159 張、噁心（Disgust）2440 張與 2562 張鄙視，本研究使用前七項情緒作為 CNN 模型訓練的資料，為 Paul Ekman 所提出的七大情緒分類。

肆、研究方法及過程

一、研究流程架構圖



圖一、研究流程結構圖（圖片來源：作者使用 Canva 繪製）

二、資料前處理

（一）灰階處理與歸一化

本研究採用了 Facial Expression Training Data（FETD）做為資料集，資料集中包含了八種情緒，憤怒、厭惡、恐懼、快樂、悲傷、驚訝、中立和鄙視，而我們則刪除了鄙視的情緒，只使用 Paul Ekman 所歸類的七種情緒作為訓練目標。而這個資料集中的影像皆為 96×96 的彩色圖像。我們會先把每個圖像轉為灰階影像，並且為了在往後加入新的特徵時造成模型過於著重某種特徵，引此我們將其進行歸一化，使其直落在[0,1]區間中。

$$x_{norm} = \frac{x - x_{min}}{x_{min} - x_{max}}$$

圖二、歸一化處理公式（圖片來源：Latex 渲染）

（二）加入干擾元素

為了提升之後模型訓練的泛化能力，我們在資料集中 70%的影像加入一些干擾的元素，包含旋轉特定角度（10 度~30 度）、加入雜訊、增加亮度（改變光照）等元素。

三、臉部的標定

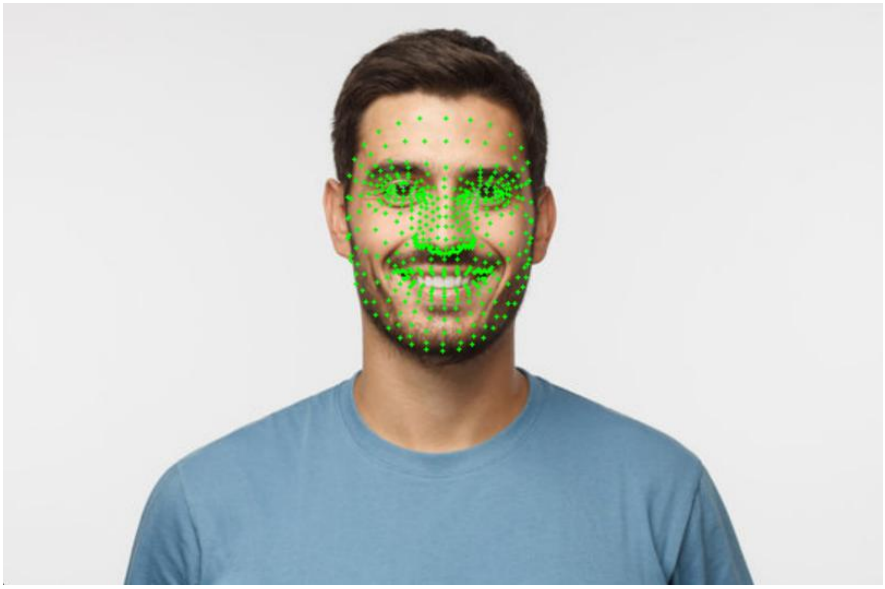
因為在實際使用時，面部位置可能不在圖片正中間，因此需要先對臉部位置進行標定，我們先使用 MediaPipe 的 FaceMesh 進行臉部特徵的標定，他可以從影像中找到臉部具有的特殊點，如圖（四）所展示，臉部共有 468 個特徵點（Landmarks）。特過這些特徵點，我們只要找到最上、最下、最左和最右邊的點就可以把臉的範圍框出來，引此我們找出 x 座標和 y 座標中最大及最小的，並向外延伸約 10 單位左右就能正確的框出臉的位置，也就是 ROI（Region of Interest），最後只要將其轉為正方形，並處理成 96×96 的灰階影像就能讓模型進行辨識。

```python
def get_face_roi(self, image, landmarks):
    h, w = image.shape[:2]
    coords = np.array([[int(l.x * w), int(l.y * h)] for l in landmarks.landmark])

    # Get face boundaries
    x_min = max(0, np.min(coords[:, 0]) - 10)
    x_max = min(w, np.max(coords[:, 0]) + 10)
    y_min = max(0, np.min(coords[:, 1]) - 10)
    y_max = min(h, np.max(coords[:, 1]) + 10)

    return image[int(y_min):int(y_max), int(x_min):int(x_max)]
```

圖三、標定臉部位置的程式碼（圖片來源：作者利用 Vscode 撰寫）



圖四、臉部地標（Landmarks）示意圖（圖片來源：素材網[9]下載後處理之結果）

四、深度學習神經網路模型的訓練

（一）模型訓練

我們的神經網絡採用卷積神經網絡來進行表情辨識。主要由卷積層（Conv）、池化層（Pooling）、全連接層（Dense）組成，每個層負責不同的任務。卷積層用來提取影像特徵，池化層則減少特徵圖大小以降低計算成本，而全連接層負責將提取的特徵轉換為對應情緒類別的可能性。我們使用 ReLU 作為激活函數與交叉熵誤差作為損失函數。

一開始我們先進行測試，找出有最佳成效的卷積層數與學習率等超參數，隨後再加入其他的臉部特徵訓練。而加入的其他特徵就可以利用前面得到的 Landmarks 座標。

對於每一次訓練，我們都會跑至其損失值約為 1 左右並且收斂時停止。

（二）模型訓練成效評估

為了評估模型在多分類情緒辨識任務中的整體效能，我們使用測試資料繪製出 7×7 的混淆矩陣，並據此計算常見的分類指標，包括 Accuracy、Precision、Recall 與 F1-score。在混淆矩陣中，對角線上的元素表示正確分類的數量 TP；每一欄中除對角線外的元素代表 FP，即其他類別預測為該類的數量；每一列中非對角線的元素則對應 FN，表示未正確預測該類別的數量；而所有不在同一列與同一欄的元素則為 TN。由於本研究中的情緒分類資料存在輕度的不平衡現象，例如「快樂」類別的樣本數量大約是「噁心」類別的 1 至 2 倍之間，若直接以 macro average 進行評估，可能會忽略常見類別的實際貢獻，導致整體指標失真。因此，我們採用 weighted average 的計算方式，根據每一類別在測試資料中的樣本數對各項指標進行加權，使資料量較少的類別能有更大的影響力。

（三）模型調整

重複步驟（一）與（二），調整並訓練，直到較佳的訓練成果，作為最後實際應用時使用的模型。

五、實際應用

我們將訓練結果最佳的模型，導入實際應用，我們決定開發一個 Google Chrome 的套件，在線上會議（即 Google Meet）中進行情緒的分析，使其能夠做為縣上課能的教學成效評估。

伍、研究結果

一、不同卷積層數（無其他特徵）的比較

（一）不使用卷積神經網路

圖五、不使用卷積神經網路的模型訓練後的 Accuracy 和 Loss 對 Epoch 作圖

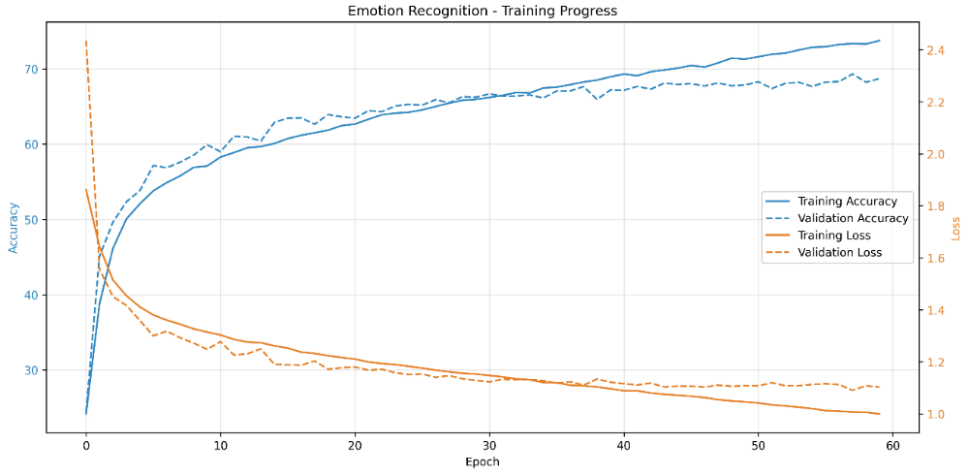
（圖片來源：作者收集數據繪製）

（二）使用三層卷積神經網路和池化層

圖六、三層卷積神經網路的模型訓練後的 Accuracy 和 Loss 對 Epoch 作圖

（圖片來源：作者收集數據繪製）

（三）使用五層卷積神經網路和池化層



圖七、五層卷積神經網路的模型訓練後的 Accuracy 和 Loss 對 Epoch 作圖

（圖片來源：作者收集數據繪製）

（四）使用七層卷積神經網路和池化層

圖八、七層卷積神經網路的模型訓練後的 Accuracy 和 Loss 對 Epoch 作圖

（圖片來源：作者收集數據繪製）

（五）結果比較與討論

我們使用了各種不同的學習率（α）進行訓練並比較後，發現當$\alpha = 0.0075$時較能夠使 Loss 值正常收斂，而上面的圖（五）、圖（六）、圖（七）以及圖（八）即為在$\alpha = 0.0075$時訓練的結果。

	無卷積層	3 層	5 層	7 層
最佳準確度	45%	65%	70%	70%
F1 分數	42	62	67	65
收斂 Epoch 數	50	45~50	40	90~100

表三、不同卷積層數的準確度與損失值（圖表來源：作者自行製作）

在準確度的部分為 5 層與 7 層最佳，因為增加特徵並不會大幅度影響輸入的維度，並且若使用 7 層作為訓練的話，訓練次數（Epoch）需要達到約 90 次時，Loss 值才會收斂至較接近 1.1（五層卷積的最終 Loss）因此我們最終決定使用下表來作為後續增加特徵後的訓練配置。

學習率α	0.0075
Batch 值	64
卷基層	5 層，每層變為原本一半（初始為 96）
其他	最大池化層、Dropout 層與全連接層

表四、訓練配置圖（圖表來源：作者自行製作）

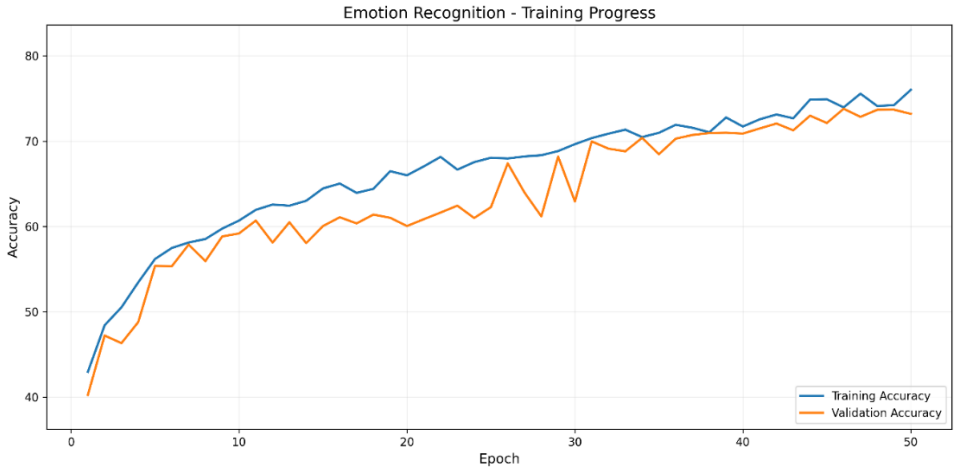
二、眉毛特徵的提取與訓練結果

有文獻提及眉毛的情況也會透露出情緒狀態，在 MediaPipe 標定的 Landmarks（臉部地標）中，有 20 個點是代表眉毛的位置，分別為左右各 10 個，我們先以鼻尖為原點，給予每個點一個新的座標。我們分別使用了三種方式做為特徵提取去訓練，根據有文獻提到的眉毛高低位置，我們測試了眉毛絕對位置與相對位置的關係，而也有文獻提到眉毛的趨勢，我們使用了曲線做為特徵。

圖九、眉毛的 20 個點（圖片來源：素材網[9]下載後處理之結果）

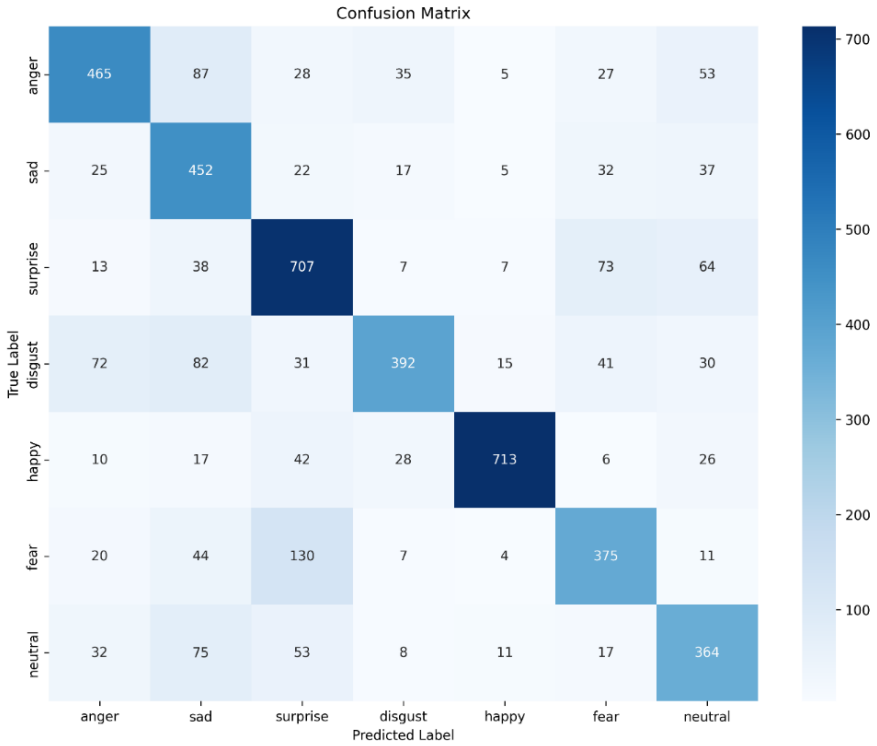
（一）使用眉毛座標作為特徵值（絕對位置）

對於這 20 個點，我們將左右眉毛各 10 個點做處理，取 x 座標相同的兩個點的終點作為輸入（左半邊眉毛第 i 個點標為L_i，右邊同理，為 R 開頭），意即增加十個輸入，為 $F = (L_1, L_2, L_3, L_4, L_5, R_1, R_2, R_3, R_4, R_5)$



圖十、採用眉毛的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）



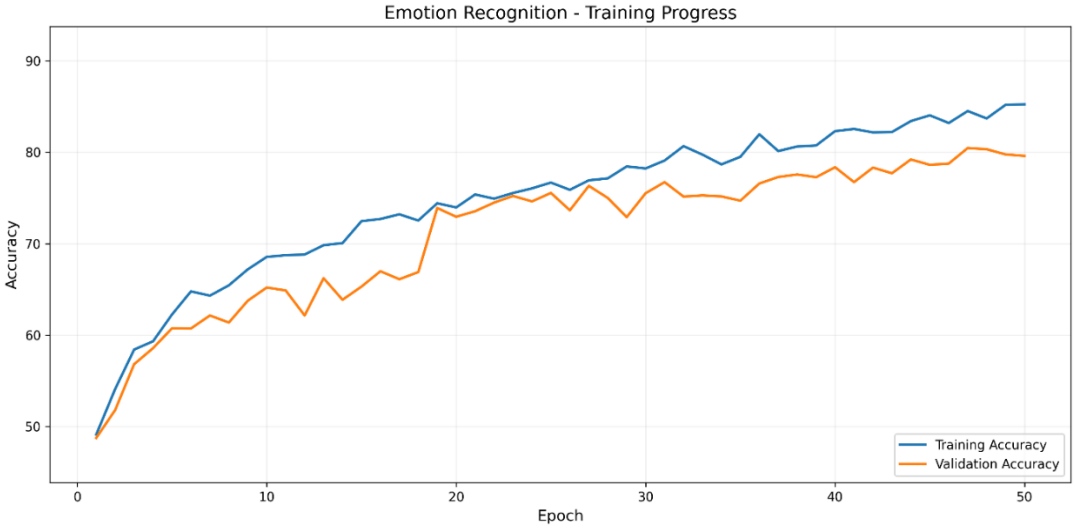
圖十一、採用眉毛的神經網路模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

（二）使用眉毛對瞳孔向量與座標作為特徵值（相對位置）

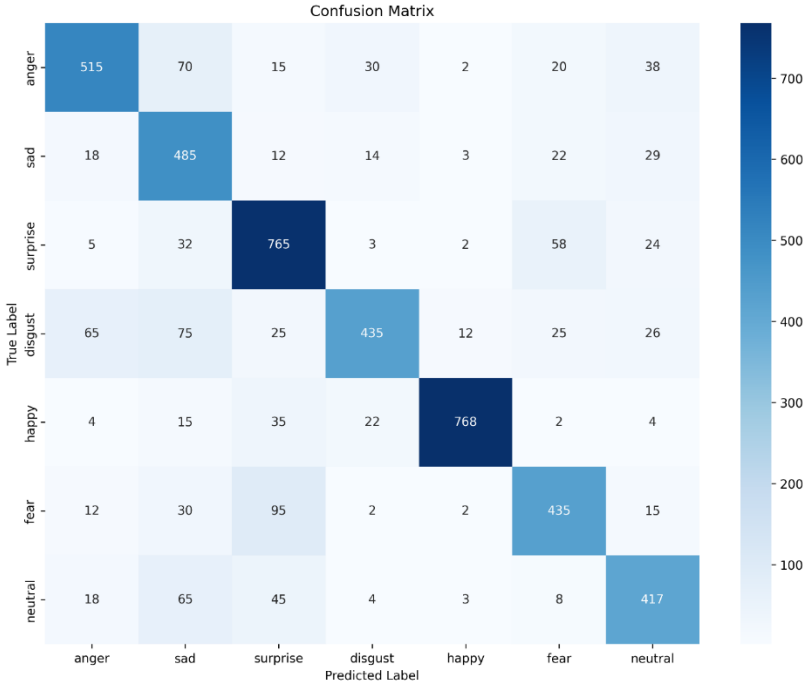
在（一）的基礎上，我們多加入了一個能代表相對位置的向量，也就是以瞳孔為基準點，對座標點點L_i或R_i做向量，得到v_{L_i}或v_{R_i}，而原本的座標，實際上也算是以鼻尖為基準點的向量，因此我們將這兩種向量作為輸入。

上述意即除了原本就有的座標點$F_1 = (L_1, L_2, L_3, L_4, L_5, R_1, R_2, R_3, R_4, R_5)$，還多了$F_2 = (v_{L1}, v_{L2}, v_{L3}, v_{L4}, v_{L5}, v_{R1}, v_{R2}, v_{R3}, v_{R4}, v_{R5})$。



圖十二、採用向量眉毛特徵的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）



圖十三、採用向量眉毛特徵的神經網路模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

（三）使用眉毛最接近四次函數擬合曲線係數作為特徵值（趨勢）

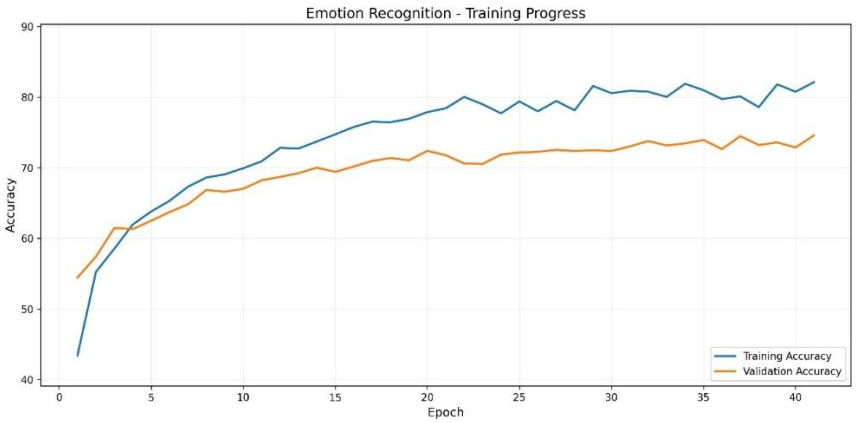
為了考慮眉毛的趨勢對情緒的影響，我們決定使用前面提到的左右眉毛分別五個中點的四次曲線係數作為特徵輸入。假設函數$Eyebrow(x) = Ax^4 + Bx^3 + Cx^2 + Dx + E$

透過五個眉毛座標點可以恰好解出 A、B、C、D、E，最後再將五個參數加入並進行訓練，即為$F = (A, B, C, D, E)$且$A, B, C, D, E \in Eyebrow(x)$

圖十四、以眉毛地標擬合四次曲線圖例（圖片來源：素材網[9]下載後處理之結果）

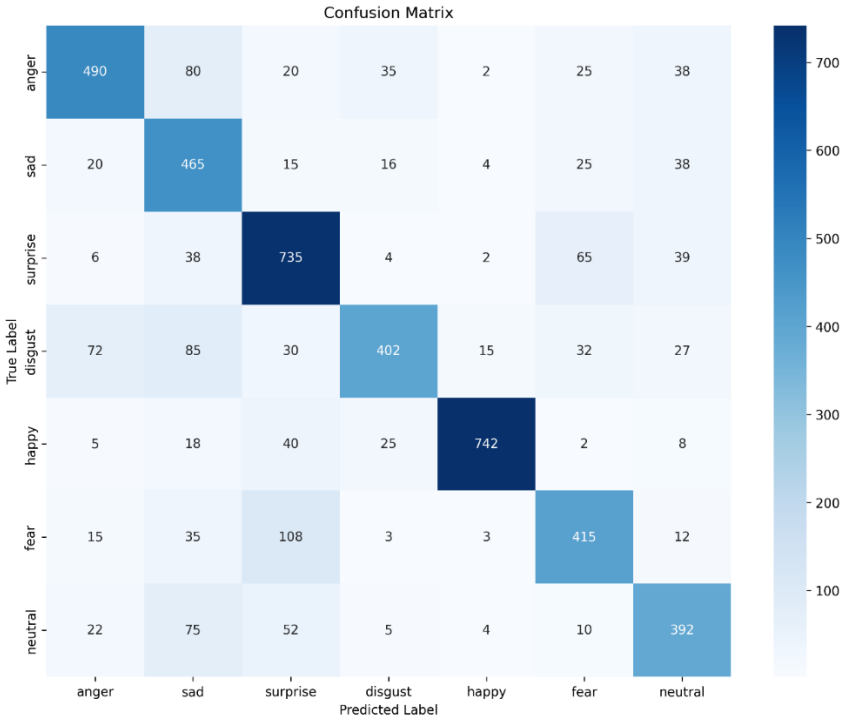
```
left: [-8.44063945e-07  8.99946891e-04 -3.42900123e-01  5.56226374e+01
 -3.13863443e+03]
right: [ 4.20375653e-06 -5.19558477e-03  2.41715100e+00 -5.01327550e+02
  3.91884556e+04]
```

圖十五、圖（十四）的曲線係數（圖片來源：作者以程式計算）



圖十六、採用眉毛擬合曲線的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）



圖十七、採用眉毛擬合曲線的神經網路模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

（四）眉毛特徵的訓練成果比較與討論（註：下列數值為七種情緒之值之算術平均）

特徵	Accuracy	Precision	Recall	F1 分數
絕對位置（鼻尖向量）	71.43%	70.02%	68.79%	67.02
相對位置（瞳孔向量）	79.15%	81.2%	78.59%	80.56
趨勢（曲線係數）	75.60%	77.15%	71.32%	74.01

表五、加入眉毛特徵的訓練成果比較表格（圖表來源：作者自行製作）

透過表（五）可以發現，不管是準確度還是 F1 分數都比沒有特徵時高出了不少。另外，以相對位置為特徵時，F1 分數能夠來到 80，代表透過瞳孔與眉毛的相對位置較能體現出眉毛位置高低對情緒狀待判別的影響，而絕對位置可能會因為每張影像中的人物差異而較不能有效區分情緒，而函數的係數可能並不能較有效的描述其趨勢，或者各種情緒間眉毛的走向趨勢對情緒狀態的影像較小。 並且於圖（十七）中可以觀察出快樂、恐懼、噁心情緒有較高的辨識準確度。

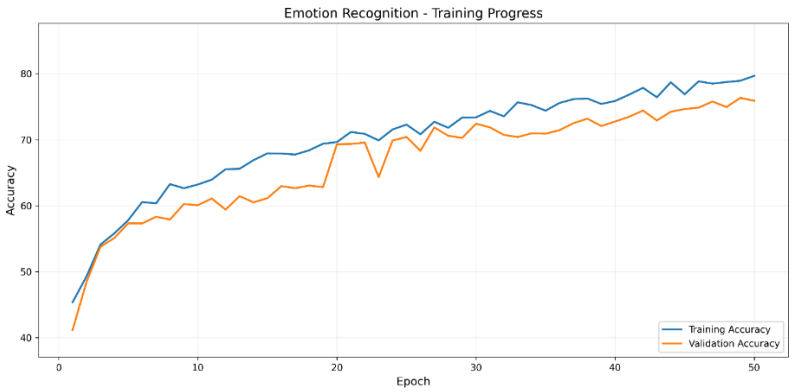
三、口部特徵的提取與訓練結果

透過 MediaPipe 提取的臉部地標可以得到 20 個座標，上下嘴唇各 10 個座標點。我們根據文獻中提到的特徵點，做了三組嘗試，分別為口部位置（絕對位置及相對位置）、口部大小（面積）。

圖十八、圍出嘴部的 20 個地標（圖片來源：素材網[9]下載後處理之結果）

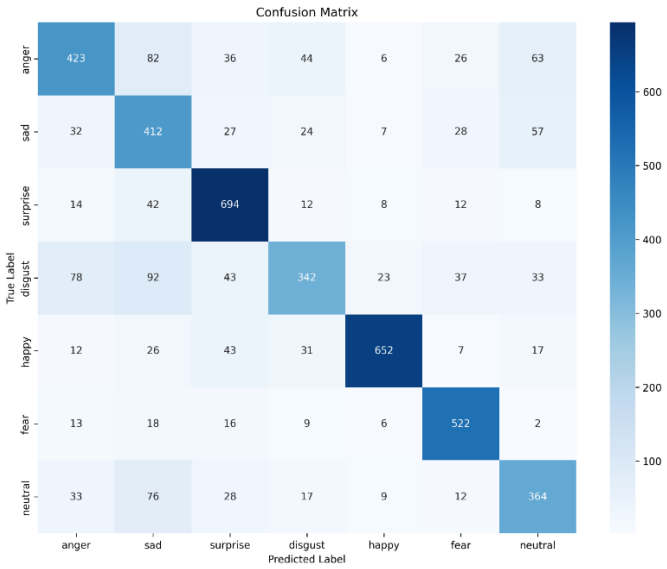
（一）將口部座標作為特徵（絕對位置）

我們先將得到的 20 個座標點直接作為輸入訓練模型。



圖十九、採用口部座標的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）

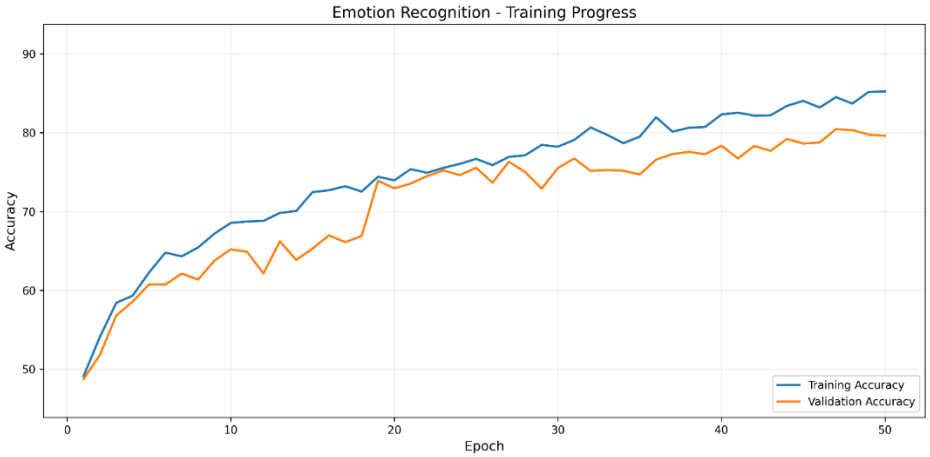


圖二十、採用口部座標的神經網路模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

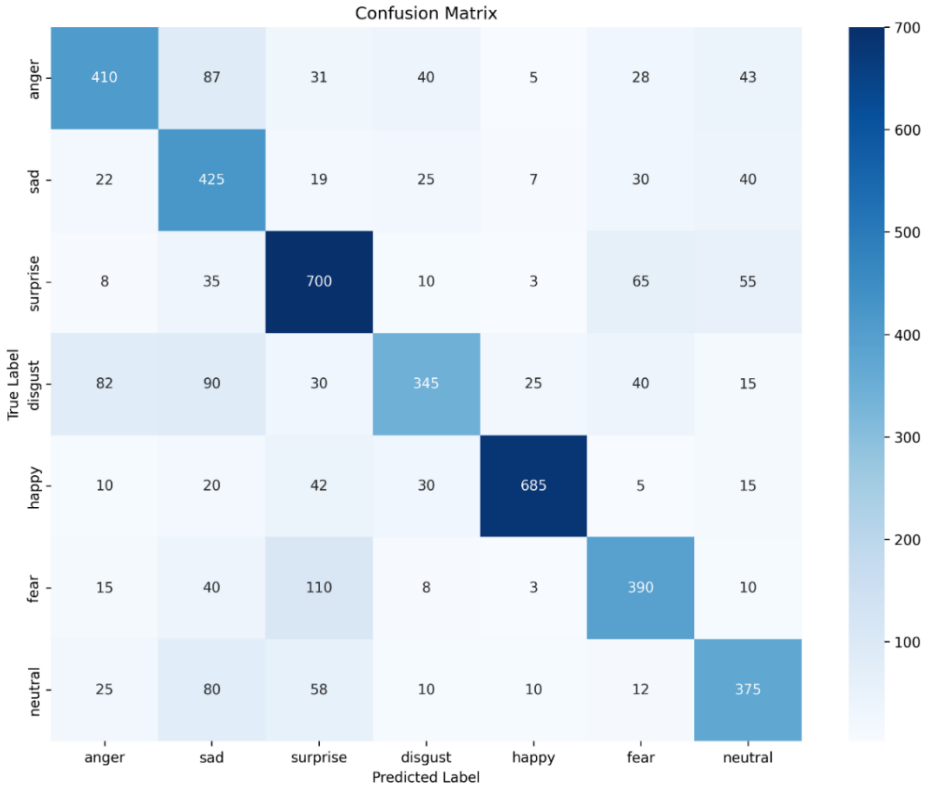
（二）將口部中點為基準點作向量（相對位置）

若以鼻尖為基準點，可能較無法展現出口部的伸張趨勢，因此在原本的基礎上將向量的基準點改為所有口部座標的幾何平均數的點，也就是口部中心的位置。



圖二十一、採用口部中點向量口部特徵的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）

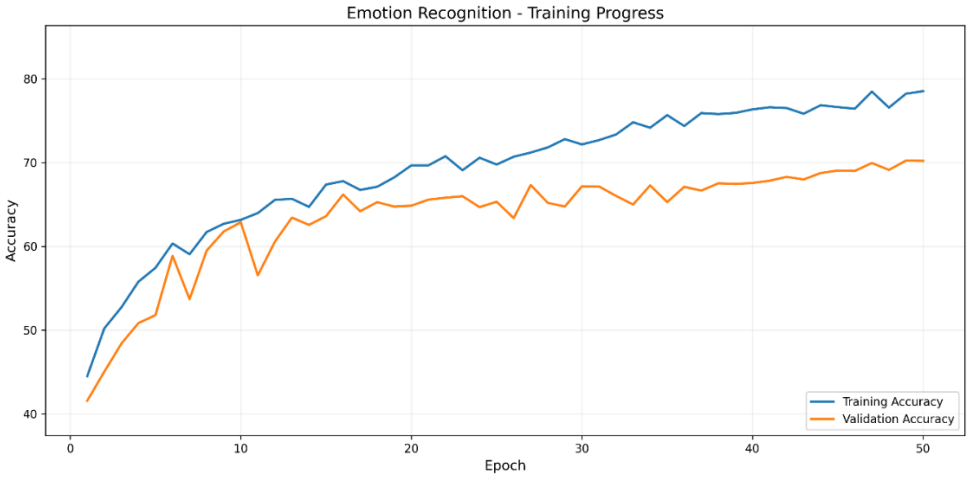


圖二十二、採用口部中點向量口部特徵的五層卷積神經網路的模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

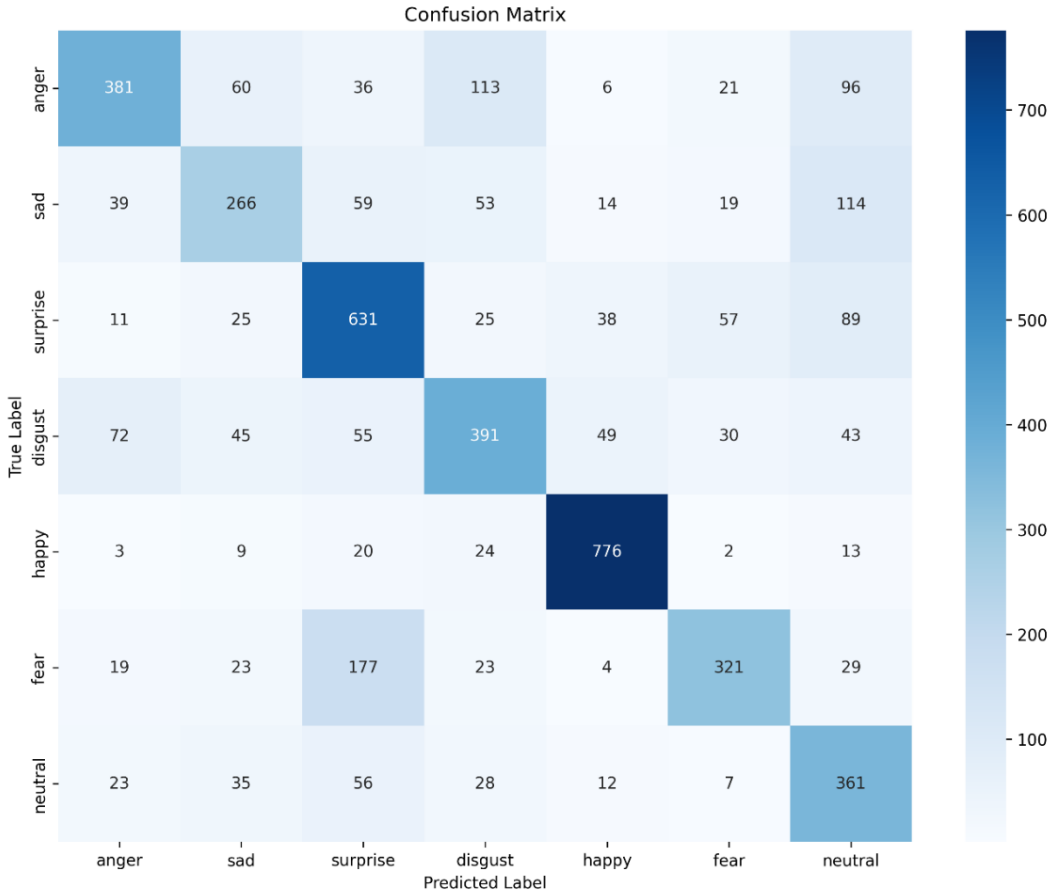
（三）將口部鼻尖向量作為特徵

為了考慮口部的大小的影響，我們計算了這 20 個點圍成的面積作為輸入。



圖二十三、採用口部面積的模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）



圖二十四、採用口部面積的模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

（四）口部特徵的訓練成果比較與討論（註：下列數值為七種情緒之值之算術平均）

特徵	Accuracy	Precision	Recall	F1 分數
絕對位置（鼻尖向量）	72.64%	74.65%	74.02%	74.36
相對位置（中心向量）	79.82%	79.14%	73.82%	77.28
口部大小（面積）	65.10%	65.15%	63.78%	64.15

表六、加入口部特徵的訓練成果比較表格（圖表來源：作者自行製作）

透過表（六）可以發現，與使用眉毛向量時相同，使用相對位置有較高的 F1 分數與準確度，我們推測其原因為相較絕對位置更能有效的表示出口部伸縮的關係，而面積的因素可能是因為大部分情緒嘴巴都不會張開，因此對情緒狀態的分析影響比較小。由圖（二十二）可看出使用口部特徵在驚訝和快樂的辨識度是比較高的，但是在其他的情緒上，很容易被便認為其他情緒。

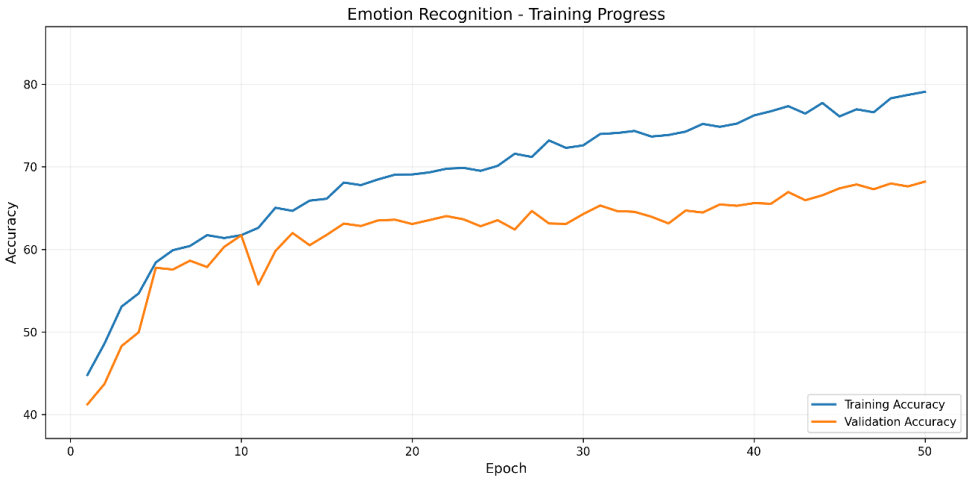
四、眼睛特徵的提取

透過 MediaPipe 可以得到 28 個代表眼睛位置的點，我們分別考慮了兩個因素的影響，一個是位置（相對位置）的影響，另一個是縮放大小的影響。

圖二十五、框出眼睛位置（圖片來源：素材網[9]下載後處理之結果）

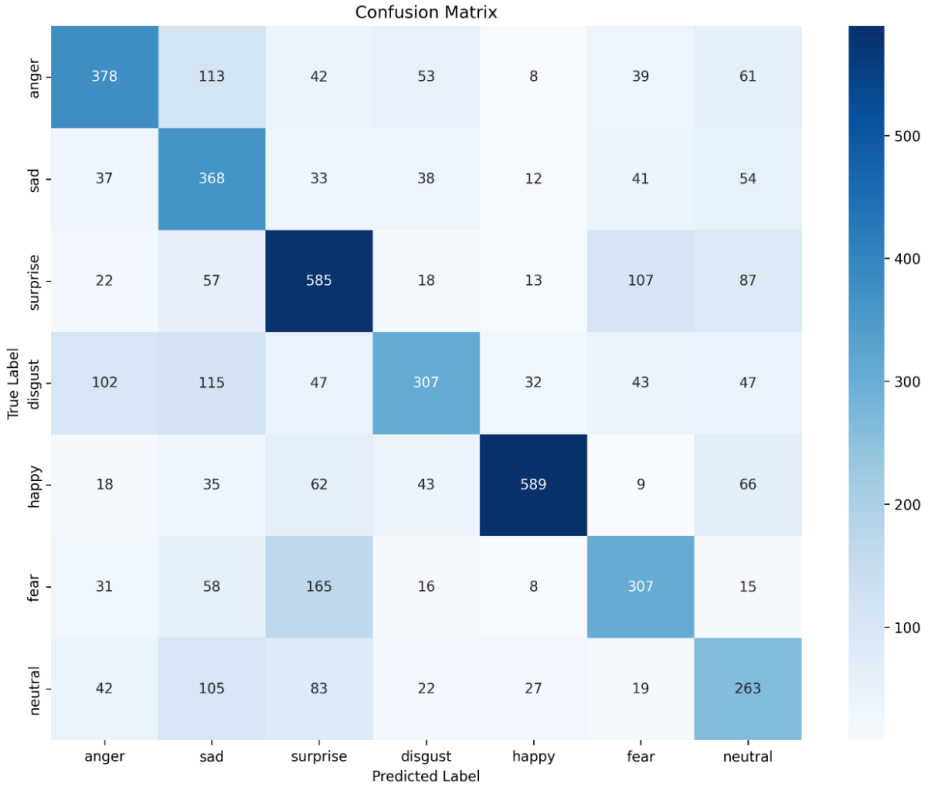
（一）將座標作為特徵（相對位置）

與口部特徵的相對位置相同，我們取這 28 個點的幾何平均中點作為基準點對這些點做向量，作為特徵訓練。



圖二十六、眼部地標的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）



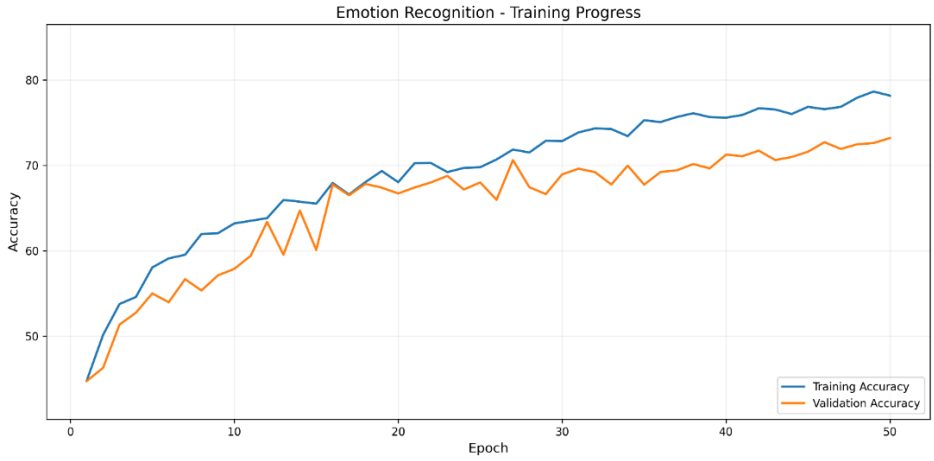
圖二十七、眼部地標的神經網路模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

（二）以眼部縮放大小為特徵

為了評估眼睛縮放大小的影響，我們考慮的 Landmarks 中演部往外一圈的點，定義其為外圈，眼部為內圈，分別計算出兩者的面積，計算其比值，作為模型輸入。即v ＝ (內圈面積,外圈面積,比值)

圖二十八、眼部與眼部外圈面積的比值（圖片來源：素材網[9]下載後處理之結果）



圖二十九、眼部外圈面積的比值的神經網路模型訓練後的 Accuracy 對 Epoch 作圖（圖片來源：作者收集數據繪製）

（三）眼部特徵的訓練成果比較與討論

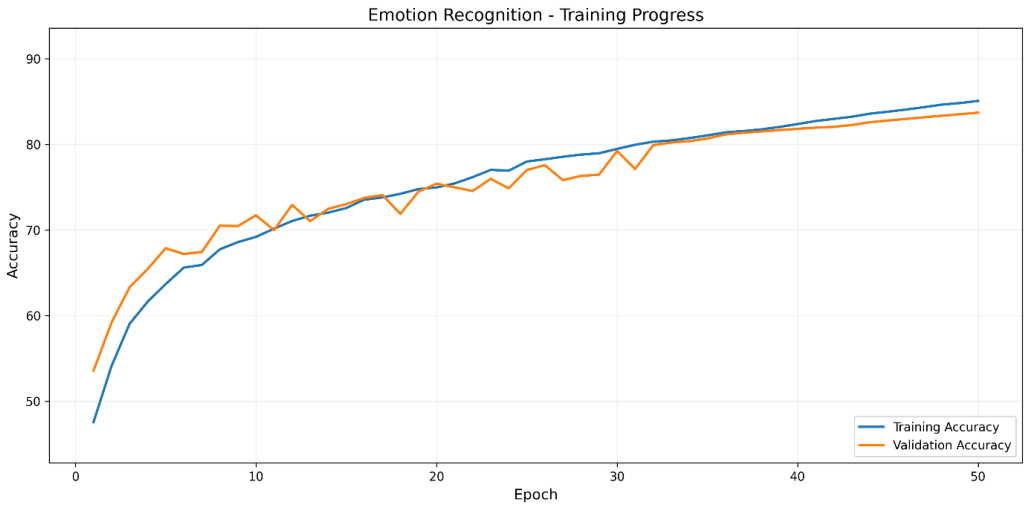
特徵	Accuracy	Precision	Recall	F1 分數
相對位置	68%	68.05%	67.72%	66
縮放大小	72%	73.14%	70.99%	71

表七、加入眼部特徵的訓練成果比較表格（圖表來源：作者自行製作）

在加入了眼部特徵後，F1 和準確率都沒有顯著的提升，我們推測相對位置和縮放比例在各種情緒中的差異不會太多，導致無法有效辨別，但是值得注意的是，在驚訝情緒和快樂情緒的準確度上是非常高的，並且在噁心的情緒上很容易被辨認為生氣或驚訝。

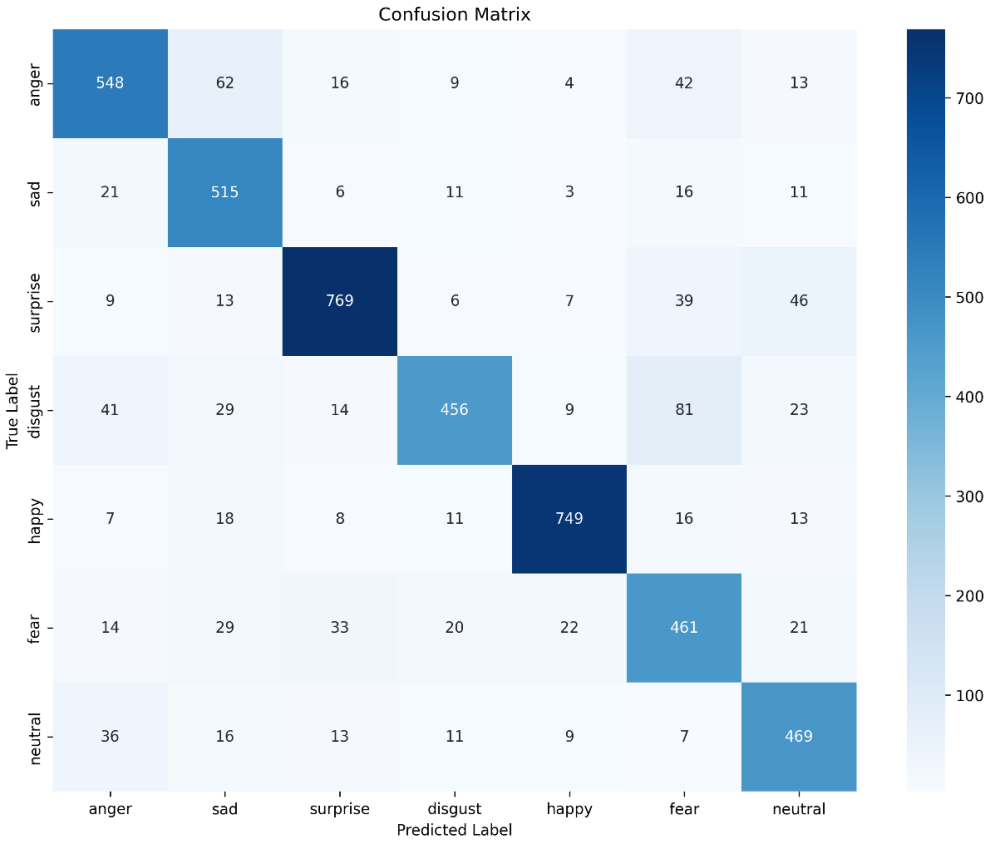
五、特徵整合訓練模型

透過了前面的各種特徵提取，發現不同特徵提取各有其較高辨識度的地方，因此我們分別取用前面每個特徵中最好的提取方式，同時套用成為最終的模型訓練，眉毛特徵取用的是瞳孔相對向量與鼻尖相對向量的混合；在口部特徵的方面則是使用口部幾何中心為基準的向量；而眼部特徵則是使用縮放大小。但是在經過幾次試驗後，發現眼部特徵的採用與否對準確度沒有明顯的差異，因此在往後的訓練中暫不採用該特徵。



圖三十、整合特徵的神經網路模型訓練後的 Accuracy 對 Epoch 作圖

（圖片來源：作者收集數據繪製）



圖三十一、整合特徵的神經網路模型訓練後的混淆矩陣

（圖片來源：作者收集數據繪製）

經過多種特徵的整合後，模型的準確率也有顯著的提升，大約為 84%，而 F1 分數也來到了前所未有的 85.6，為目前最佳，並且透過圖（三十一）可以看出基本上大部分情緒皆有相當高的辨識準確度。

從對角線上的數值可以看出，快樂與驚訝情緒的辨識準確度非常高，對角線上的數值顯著大於其他情緒。這與研究結果中提到的，不管是加入眉毛或口部特徵，快樂情緒的個別 F1 分數都高於 90 相互呼應，顯示模型對正向情緒的接受判斷能力很強。相較之下，憤怒、悲傷和 厭惡情緒在對角線上的數值似乎相對較低，可能是「表情細部特徵的差異較小或難以量化」有關。

透過整合眉毛和口部兩種特徵後，所訓練出的模型在七種情緒辨識上的最終表現。它證明了整合不同面部區域的特徵確實能提升模型的整體辨識能力，尤其在快樂、驚訝和恐懼這幾種情緒上表現非常出色。同時，混淆矩陣也清晰地指出了模型的不足...

六、模型應用與成果

在訓練完模型後，我們將其與 Google Meet 做結合

（一）用戶影像獲取

在獲取影像這方面，我們決定自製一個 Chrome 插件取得 Google Meet 頁中的 <video>，其中包含了各個參與者的視訊畫面，可以在此擷取出稍後需要的影像。

```
videoElement = document.querySelectorAll("video")
function captureVideoFrame(videoElement) {
    const canvas = document.createElement("canvas");
    canvas.width = videoElement.videoWidth;
    canvas.height = videoElement.videoHeight;
    const ctx = canvas.getContext("2d");
    ctx.drawImage(videoElement, 0, 0, canvas.width, canvas.height);
    return canvas.toDataURL("image/png");
}
```

圖三十二、以插件獲取影像的範例（圖片來源：作者使用 Vscode 撰寫）

（二）模型預測

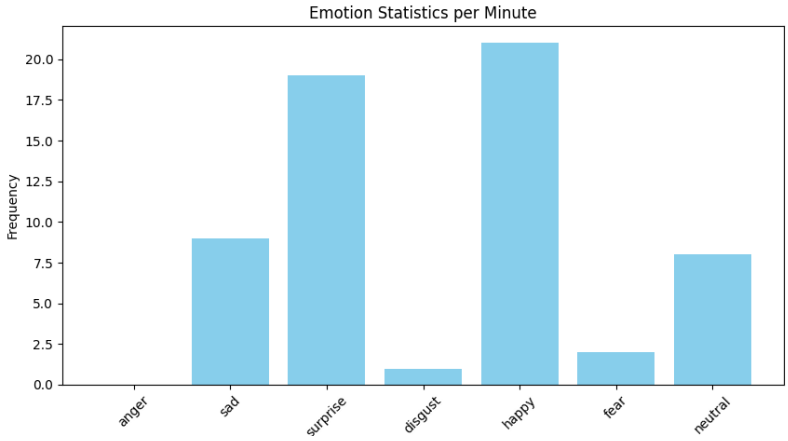
在擷取出影像後，將其交給之前訓練完的模型預測當前的情緒，只記錄機率最高的答案，並且收集一段時間內的預測結果用於分析。

（三）圖表化結果與網頁回饋

最後輸出圖表，目前採用的是每秒一張圖片，每一分鐘統計一次，再使用伺服器功能回饋預測結果給 Google Meet 網頁方便檢查是否每個步驟都順利執行。

（四）製作 API

為了方便使用，以 Gradio 和 Flask 製作簡易的服務接口（API），並且將原本的程式套入其中，再加上調整表格格式與單張圖像辨識等功能。



圖三十三、圖表畫結果之範例（影像來源：程式碼統計後產生）

圖三十四、網頁回饋之範例（影像來源：由實際畫面除去作者訊息）

圖三十五、簡易 API 之介面（影像來源：由作者擷取實際螢幕畫面）

陸、討論

本研究目前初步展現了在情緒狀態分析上的潛力，目前使用此系統能夠有效分析受試者的情緒狀態，特別是在快樂、恐懼和驚訝等情緒的辨識上準確度相當高。不過，對於其他情緒的辨識準確度仍有待提升，這可能與其他表情細部特徵的差異較小或難以量化有關，進而使模型在捕捉這些微妙變化時出現困難。 此外，還可能存在其他因素影響準確度。首先，不同受試者即便表現相同情緒，其面部肌肉的動作幅度與速度也可能存在差異，這種個體間的多樣性會導致模型難以從有限的訓練資料中涵蓋所有可能的變化。其次，外部環境因素也不容忽視。例如，光線條件、拍攝角度以及背景雜訊都可能干擾面部特徵的清晰呈現，使得模型在分析時難以獲得完整的情緒資訊。另外，目前的特徵提取演算法可能依賴明顯且易於辨識的臉部區域，對於那些隱藏在微表情變化中的細節則可能無法充分捕捉，這也是導致部分情緒準確度較低的重要原因。最後，現有模型在設計上主要針對無其他配件情況下的臉部進行辨識，因此一旦受試者佩戴口罩或墨鏡，關鍵面部特徵區域便會部分或完全遮擋，這不僅會引起誤判，還可能使輸入特徵資料產生偏差。未來若能引入更多考慮遮擋情況的資料訓練，或利用其他技術改善特徵擷取，則有助於提升模型在各種情況下的泛化能力與辨識準確度。

柒、結論

一、深度學習神經網路模型訓練之成效

不管是加入眉毛特徵或是口部特徵，其快樂情緒個別 F1 分數皆高於 90，表示對於正向情緒的接受判斷能力是非常好的。在眉毛特徵中，除了快樂情緒，恐懼情緒的辨識度也相當不錯；在口部特徵中，驚訝情緒也有著與快樂情緒同樣優異的 F1 分數。最後統合兩種特徵後，其準確度達到，F1 分數為 84，為本研究最佳之成效，也最為後續實際應用之模型。

二、實際應用層面之討論

目前以實作出於 Google Meet 中即時紀錄情緒並統計分析繪製成長條圖的 Chrome 插件，達到了我們預期的效果，顯示了此方法與技術的可行性。能夠作為老師在線上課程中對於學生成效評估的輔助工具。

透過標準化的服務接口（API）對外提供，能更加發揮其應用的潛力。此舉旨在讓心理諮詢、醫療診斷及智能系統開發等領域，能更輕易結合本模型，加速其在多元創新場景的落地與推廣，實現以客觀情緒分析作為評估工具的初衷。

四、未來展望

（一）進一步提升模型的準確度

（二）強化各種特徵的提取

（三）擴充資料集加強不同狀況的處理，像是戴眼鏡或者戴墨鏡

（四）應用於其他領域，例如心理諮詢、醫療診斷或開發情緒感知的智能系統

捌、參考文獻資料

[1] Alizadeh, S., & Fazel, A. (n.d.). *Convolutional neural networks for facial expression recognition*. Stanford University.

[2] Li, S., & Deng, W. (2018). *Deep facial expression recognition: A survey*. Beijing University of Posts and Telecommunications.

[3] Hosseini, M., Bodaghi, M., Bhupatiraju, R. T., Maida, A., & Gottumukkala, R. (2023). *Multimodal stress detection using facial landmarks and biometric signals*. University of Louisiana at Lafayette.

[4] 孔令琴, 陳飛, 趙躍進, 董立泉, 劉明, & 惠梅. (2021). *融合心率變異性與表情的非接觸心理壓力檢測*. 光學學報, 41(3), 0310003.

[5] Patinge, S., Dhanwari, A., Lode, R., Nakhate, S., Biswas, G., & Hatwar, N. (2020). *Stress detection using facial expression*. International Journal of Emerging Trends in Engineering and Basic Sciences, 7(Special Issue 2), 256-260.

[6] Dukić, D., & Sovic Krzic, A. (2022). *Real-time facial expression recognition using deep learning with application in the active classroom environment*. Electronics, 11(8), 1240.

[7] Narayana, S., Jain, S., Katti, H., Goecke, R., & Subramanian, R. (2022, July 18). *Affective computational advertising based on perceptual metrics*. University of Canberra.

[8] 顏乃欣 (2010)。*情緒對決策歷程的影響*. 人文與社會科學簡訊, 11(4), 113-120。

[9] Adobe Stock https://reurl.cc/EVAyDK

【評語】052510

此作品以深度學習神經網路模型進行面部表情的辨識,利用現有的公開資料及訓練,從影像辨識人臉影像的七種可能情緒的狀態。此作品嘗試加入經由臉部特徵點抽取眉毛、口部與眼睛等多部位特徵萃取以提升其精確度。作者整合此模型至 Google Meet 應用程式，作品具完整開發流程與實際應用針對性。

未來發展建議：

1. 若能整合即時語音與影像之表情多模態分析應可增強其表情辨識精確度，建議可增加 Voice Emotion Recognition 模組進行整合。

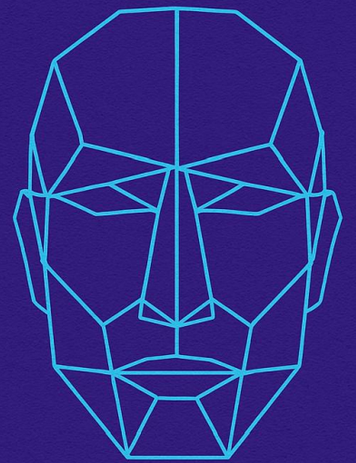
2. 目前模型受人臉影像解析度不足影響甚大，應考慮可獲得較高影像解析度之實際應用情境。

3. 建議可考慮與其他更適合的應用結合，如 AI 助教或智慧機器人。

作品海報

全「面」分析：

基於面部表情的情緒模型



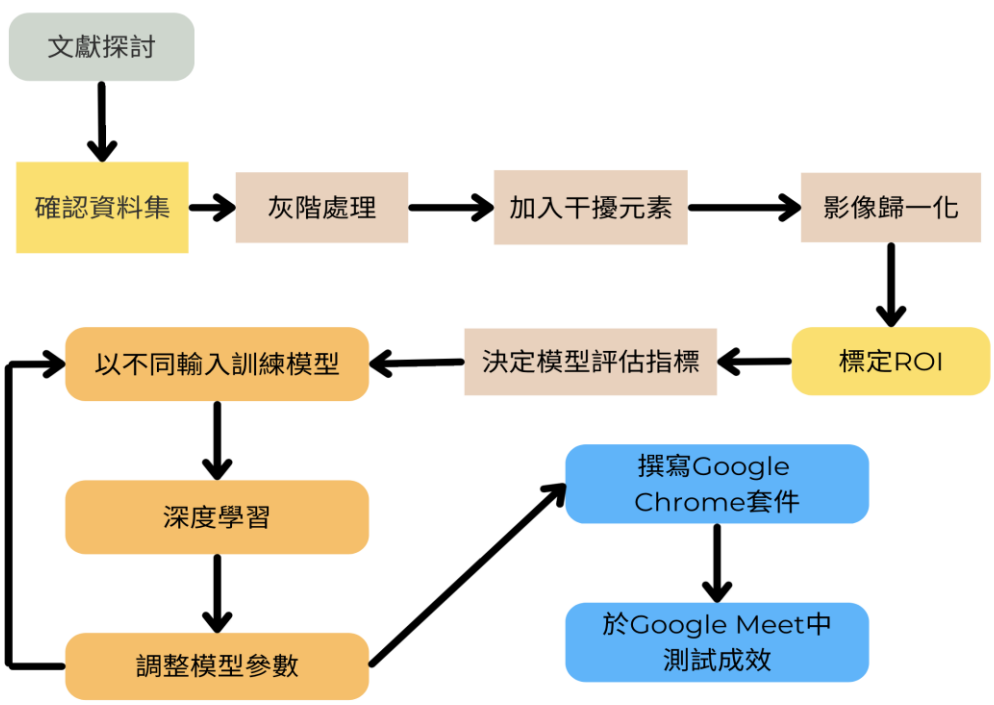
摘要

本研究結合MediaPipe與深度學習模型，從臉部影像中提取眉毛、口部、眼部等關鍵特徵進行情緒辨識訓練，最終整合多項特徵後，模型準確率達84%、F1分數為85.6。研究亦成功開發可應用於Google Meet的即時情緒分析插件，協助遠端會議中理解參與者情緒反應，展現面部表情辨識於實務中的可行性與應用潛力。

壹、研究目的

一、應用Mediapipe準確擷取臉部影像。
二、利用深度學習神經網路框架訓練出成效好的評估模型。
三、面部表情辨識的模型可以準確的辨識出使用者的面部表情。
四、面部表情辨識模型的實際應用。

貳、研究方法及過程



研究流程圖（作者自行繪製）

一、使用資料集

Facial Expressions Training Data from Kaggle

anger	disgust	fear	happy	neutral	sad	surprise
3132	2440	3159	4524	4508	3020	3997

此分類由Paul Ekman提出，並在多數參考的論文中皆被提及，因此本實驗採取同樣的分類模式。

二、資料前處理

原資料集中的圖像為96×96的彩色圖像，我們會先對其進行灰階處理，將其進行歸一化。為了提升之後模型訓練的泛化能力，我們在資料集中70%的影像加入干擾元素，包含旋轉特定角度（10度~30度）、調整亮度、對比度、飽和度等（隨機調整至80%～120%）。

參、研究結果

一、不同卷積層數（無其他特徵）的比較

	無卷積層	3層	5層	7層
最佳準確度	45%	65%	70%	70%
F1分數	42	62	67	65
收斂Epoch數	50	45~50	40	90~100

表一、不同卷積層數的準確度與損失值（來源作者自行製作）

在準確度的部分為5層與7層最佳，因為增加特徵並不會大幅度影響輸入的維度，並且若使用7層作為訓練的話，訓練次數（Epoch）需要達到約90次時，Loss值才會收斂至較接近1.1（五層卷積的最終Loss）因此最終決定使用下表來作為後續增加特徵後的訓練配置。

學習率α	0.0075
Batch值	64
卷積層	5層，每層變為原本一半（初始為96）
其他	最大池化層、Dropout層與全連接層

表二、訓練配圖（來源作者自行製作）

二、眉毛特徵的提取與訓練結果

有文獻提及眉毛的情況也會透露出情緒狀態，在 MediaPipe 標定的 Landmarks（臉部地標）中，有 20 個點是代表眉毛的位置，分別為左右各 10 個，我們先以鼻尖為原點，給予每個點一個新的座標。我們分別使用了三種方式做為特徵提取去訓練，根據有文獻提到的眉毛高低位置，我們測試了眉毛絕對位置與相對位置的關係，而也有文獻提到眉毛的趨勢，我們使用了曲線做為特徵。

特徵	Accuracy	Precision	Recall	F1 分數
絕對位置（鼻尖向量）	71.43%	70.02%	68.79%	67.02
相對位置（瞳孔向量）	79.15%	81.2%	78.59%	80.56
趨勢（曲線係數）	75.60%	77.15%	71.32%	74.01

表三、加入眉毛特徵的訓練成果比較表格（來源作者自行製作）

圖一、眉毛的 20 個點
（取自 Khabirov, n.d.，經本研究處理）

三、口部特徵的提取與訓練結果

我們透過 MediaPipe 提取的臉部地標可以得到 20 個座標，上下嘴唇各 10 個座標點。我們根據文獻中提到的特徵點，做了三組嘗試，分別為口部位置（絕對位置及相對位置）、口部大小（面積）。

特徵	Accuracy	Precision	Recall	F1 分數
絕對位置（鼻尖向量）	72.64%	74.65%	74.02%	74.36
相對位置（中心向量）	79.82%	79.14%	73.82%	77.28
口部大小（面積）	65.10%	65.15%	63.78%	64.15

表四、加入口部特徵的訓練成果比較表格（來源作者自行製作）

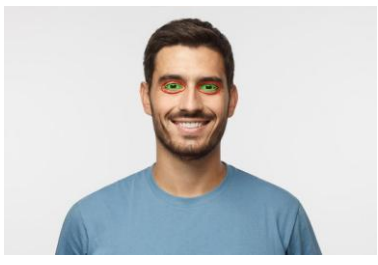
圖二、圍出嘴部的 20 個地標
（取自 Khabirov, n.d.，經本研究處理）

四、眼睛特徵的提取

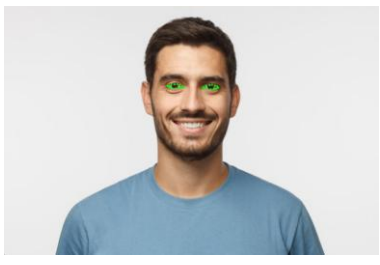
透過 MediaPipe 可以得到 28 個代表眼睛位置的點，我們分別考慮了兩個因素的影響，一個是點與點間位置（相對位置）的影響，另一個是縮放大小的影響，即眼部外圈，內圈的面積比值。

特徵	Accuracy	Precision	Recall	F1 分數
相對位置	68%	68.05%	67.72%	66
縮放大小	72%	73.14%	70.99%	71

表五、加入眼部特徵的訓練成果比較表格（來源作者自行製作）



圖三、框出眼睛位置

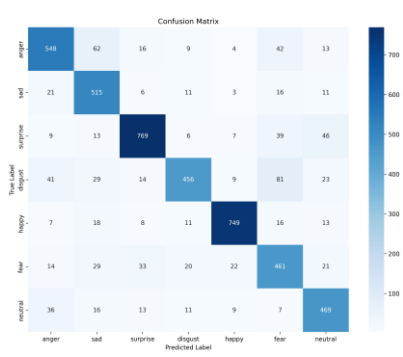


圖四、眼部與眼部外圈面積
（取自 Khabirov, n.d.，經本研究處理）

肆、討論

一、特徵整合訓練模型

透過了前面的各種特徵提取，並且不同特徵提取各有其較高辨識度的地方，因此我們選取個特徵中較高辨識度的取用方式，同時套用來作為最終的模型訓練，眉毛特徵取用的是瞳孔相對向量與鼻尖相對向量的混合；在口部特徵的方面則是使用口部幾何中心為基準量；而眼部特徵經試驗後發現成效較為普通而不採用。



圖五、整合特徵的神經網路模型訓練之成果（來源作者自行製作）

Accuracy	Precision
82%	83.15%
Recall	F1 分數
82.61%	85.2

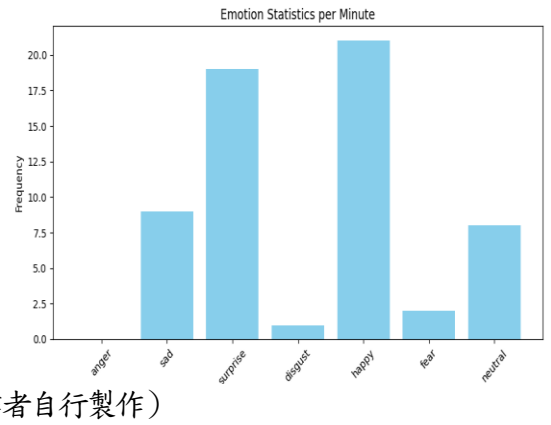
表六、模型訓練之成果（來源作者自行製作）

經過多種特徵的整合後，模型的準確率也有顯著的提升，大約為 82%，而 F1 分數也來到 85.2，可以看出基本上大部分情緒皆有相當高的辨識準確度，我們推測可能是因為採用了兩種特徵，使其原本差異不大的特徵多了一項評斷標準，故能有效地增加準確度，但 disgust 與 fear 的準確率仍低於其他表情。

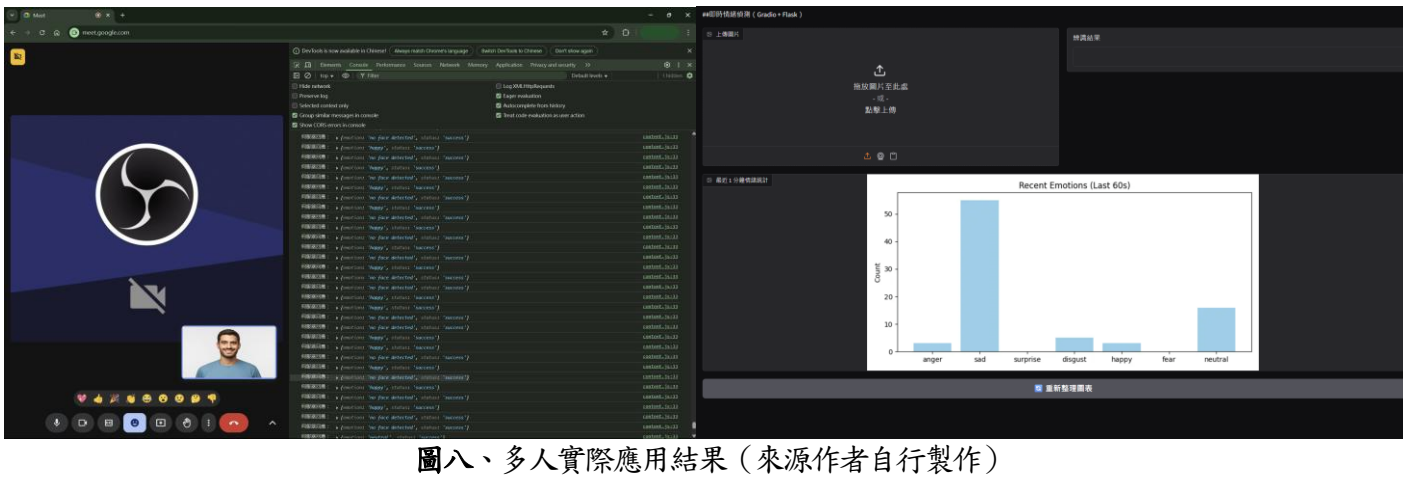
二、模型應用與成果

在訓練完模型後，我們將其與 Google Meet 做結合，做法是自製一個 Chrome 插件，Chrome 插件會先取得 Google Meet 網頁中各用戶的<video>串流資料，裡面包含了用戶的視訊畫面，再將圖像傳輸到本機端後交給模型判斷，將傳輸是否成功以及判斷的結果傳回網頁，最後統計一段時間內出現的表情比例並表格化輸出。

圖六、<video>串流資料範例（由作者擷取）



圖七、單人測試結果（來源作者自行製作）



圖八、多人實際應用結果（來源作者自行製作）

在使用 Google Meet 時加入本模型的應用，可以即時以表格的形式，讓使用者得知當前會議中用戶對於內容的大致反饋，可以協助調整內容或製作成果報告。

三、未來展望

（一）未來改進方向

部分情緒辨識準確度偏低

快樂、恐懼、驚訝辨識效果佳；其他情緒辨識仍待提升。

特徵提取演算法侷限

目前依賴明顯的臉部區域，對微表情變化的細節掌握不足。

受遮擋影響辨識準確性

佩戴口罩或墨鏡會遮擋關鍵面部特徵，導致辨識誤差或資料偏差。

缺乏針對遮擋情境的訓練資料

模型泛化能力有限，難以應對真實場景中的多樣化遮擋狀況。

（二）更多應用可能

心理諮詢輔助工具

應用於情緒狀態即時監測與判讀，輔助心理專業人員。

公共廣告反應統計

設置於廣告牌旁，統計人群情緒反應，提高行銷效果分析。

車內駕駛狀態監控

分析駕駛情緒穩定度，預防危險駕駛行為或疲勞駕駛。

多情境辨識能力的強化

若提升泛化能力，未來可應用於監視、客服、教育等多種場域。

肆、結論

不管是加入眉毛特徵或是口部特徵，其快樂情緒個別 F1 分數皆高於 90，表示對於正向情緒的接受判斷能力是非常好的。在眉毛特徵中，除了快樂情緒，恐懼情緒的辨識度也相當不錯；在口部特徵中，驚訝情緒也有著與快樂情緒同樣約莫 89 的 F1 分數。最後統合兩種特徵後，其準確度達到 82%，F1 分數為 85，為本研究最佳之成效，也最為後續實際應用之模型。目前以實作出於 Google Meet 中即時紀錄情緒並統計分析並結合 API 介面繪製成長條圖的 Chrome 插件，達到了我們預期的效果，顯示了此方法與技術的可行性。能夠作為視訊主講者，觀察遠端受眾的表情回饋的輔助工具，並且將來應用於更多領域。