

2024 年【科學探究競賽-這樣教我就懂】

普高組 成果報告表單

題目名稱：面具背後的情感

一、摘要

疫情下戴口罩導致難以觀察他人面部表情，影響了情緒交流，尤其在非面對面場合更為明顯。因此，我們計劃研究如何運用面部辨識技術，通過分析特定特徵點，開發一套能夠識別口罩下情緒的系統。這將有助於解決人際互動障礙，同時也可在社會安全（如防止詐騙）和娛樂（如人臉特效）方面發揮作用。

二、探究題目與動機

近三年來，即使台灣已經取消了強制戴口罩的規定，但戴口罩已經成為部分人的日常習慣。然而，口罩遮蔽了面部大部分特徵，導致情緒表達困難。情緒在人際交往中至關重要，但面部辨識技術的進步提供了一種可能性。配合情緒分析，這種技術能在公共安全、醫療等方面提供幫助，如增加遠程溝通的準確性，提前發現潛在犯罪行為等。因此，研究口罩下的情緒分析並開發相應系統具有實際意義。

三、探究目的與假設

- 1.了解情緒辨識之原理，探討不同情緒辨識的種類、功能、以及應用
- 2.建立開發環境並實際操作辨識情緒系統
- 3.結合口罩判斷與戴口罩情況下之情緒辨識
- 4.比較並分析市面上情緒辨識系統的差異及優缺點
- 5.統整開發系統及未來前景

四、探究方法與驗證步驟

研究透過整合 Dlib 和 OpenCV，我們將實際捕捉人臉，分析人臉特徵，並進行情緒辨識。這兩種方法將提供多角度的觀點，以更全面地理解口罩下的情緒辨識挑戰與可能解決方案。

（一）系統實作流程

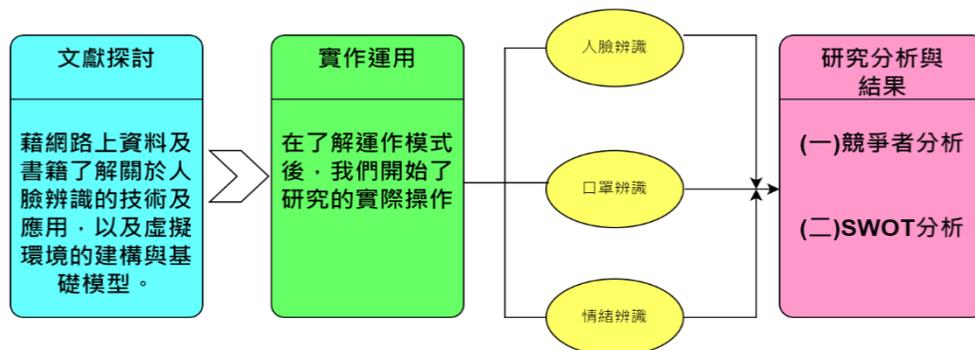


圖 1：研究流程圖

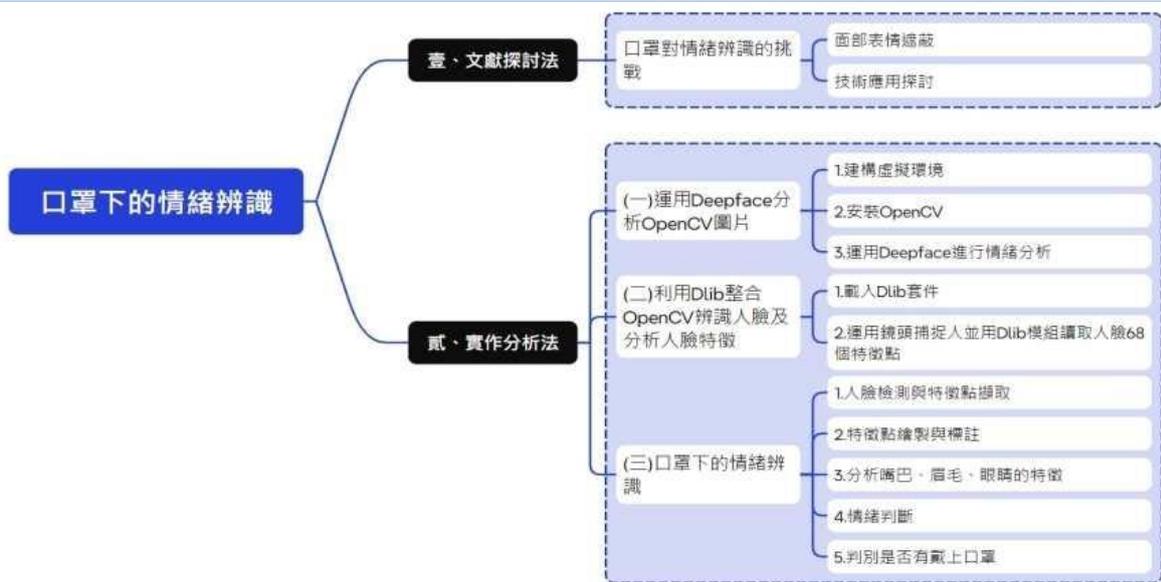


圖 2：研究架構圖

(二) 建置環境

1、載入 Dlib 套件

Dlib 為開源機器學習工具套件，特別是計算機視覺和機器學習領域。它提供了我們豐富的功能，人臉檢測、人臉識別、臉部關鍵點檢測等人臉相關的任務。高度效能的資料庫，能夠在多種平台上運行（例如我們使用的 Python），並且有豐富的文檔和示例代碼。

(1) pip install cmake-簡化使用 CMake（一個開源的構建工具）的 Python 項目的過程。安裝 cmake 後，可以在 Python 代碼中使用，以更方便地配置和構建 CMake 項目。這適用於在 Python 中進行跨語言開發，如圖 3。

```
C:\Users\aa162>pip install cmake
Collecting cmake
  Obtaining dependency information for cmake from https://files.pythonhosted.org/packages/d4/f3/cc82bb2ef5e9d3396007a18c
e2c4b5fa57d37ff95e36174b478df6fc0918/cmake-3.27.5-py2.py3-none-win_amd64.whl.metadata
  Downloading cmake-3.27.5-py2.py3-none-win_amd64.whl.metadata (6.8 kB)
  Downloading cmake-3.27.5-py2.py3-none-win_amd64.whl (34.6 MB)
  34.6/34.6 MB 4.2 MB/s eta 0:00:00
Installing collected packages: cmake
Successfully installed cmake-3.27.5
```

圖 3 資料來源：研究者電腦自行截圖

(2) 在官網下載 Dlib 安裝包，解壓縮後放入 Python 資料夾內

2、運用鏡頭捕捉人臉並用 Dlib 模組分析 68 點特徵

檔案名稱	日期	時間	大小
instance_segmentation_voc2012net_v2.dnn	2020-01-20	05:56	24M
mmod_dog_hipsterizer.dat.bz2	2016-10-02	10:02	17M
mmod_front_and_rear_end_vehicle_detector.dat.bz2	2017-09-11	04:21	3.7M
mmod_human_face_detector.dat.bz2	2016-10-07	19:01	678K
mmod_rear_end_vehicle_detector.dat.bz2	2017-08-25	15:32	3.6M
resnet34_1000_imagenet_classifier.dnn.bz2	2016-06-25	18:54	79M
resnet50_1000_imagenet_classifier.dnn.bz2	2020-02-07	06:05	83M
semantic_segmentation_voc2012net.dnn	2017-12-18	07:29	148M
semantic_segmentation_voc2012net_v2.dnn	2019-01-06	07:04	153M
shape_predictor_5_face_landmarks.dat.bz2	2017-09-15	16:51	5.4M
shape_predictor_68_face_landmarks.dat.bz2	2015-07-24	05:19	61M



圖 4：在安裝資料夾中找到目標檔案-Dlib 人臉的 68 的特徵點

到 Dlib 函式庫下載 68 個特徵點的檔案模型解壓縮後放入程式跟目錄，利用 dlib 供訓練好的模型，可以辨識出人臉的 68 的特徵點，68 特徵點包括鼻子、眼睛、眉毛，以及嘴巴等等，如上圖 4。

(三) 情緒辨識

我們先利用 68 個特徵點的檔案模型標示出人臉的大小，再使用圓圈標示出 68 個特徵點並進行編碼，以便之後的情緒分析。

1、人臉檢測與特徵點擷取

首先，透過 OpenCV 中的人臉檢測器，我們能夠檢測圖像中的人臉。對於每個檢測到的人臉，我們使用 enumerate 遍歷所有的人臉，並使用 cv2.rectangle 繪製一個紅色的矩形框以標記人臉的位置。接下來，使用 68 個特徵點的預測器 (self.predictor)，得到每個特徵點的 (x,y) 坐標，如圖 5。

```
for k, d in enumerate(faces):
    cv2.rectangle(im_rd, (d.left(), d.top()), (d.right(),
d.bottom()), (0,0,255))
    shape = self.predictor(im_rd, d)
```

圖 5：取得特徵點座標之程式碼，於 VS Code 撰寫程式之截圖

2、特徵點繪製與標註

在每個人臉的特徵點上繪製綠色圓圈，並在圓圈旁邊標註特徵點的編號。這使我們能夠視覺化每個特徵點的位置，如圖 6。

```
for i in range(68):
cv2.circle(im_rd, (shape.part(i).x, shape.part(i).y), 5, (0,255,0), -1,
8)
cv2.putText(im_rd, str(i), (shape.part(i).x, shape.part(i).y),
```

圖 6：在特徵點上繪製綠圈之程式碼，於 VS Code 撰寫程式之截圖

3、分析嘴巴、眉毛和眼睛的特徵

(1) 嘴巴：計算嘴巴的寬度和高度，以及嘴巴的張開程度。這些值將用於後續的情緒判斷，如圖 7。

```
mouth_width = (shape.part(54).x - shape.part(48).x) /
self.face_width
mouth_height = (shape.part(66).y - shape.part(62).y) /
self.face_width
```

圖 7：計算嘴巴的張開程度之程式碼，於 VS Code 撰寫程式之截圖

(2) 眉毛：通過分析眉毛上的特徵點，計算眉毛的高度和兩側的距離。同時，擬合一條直線，計算眉毛的傾斜程度。

(3) 眼睛：計算眼睛的睜開程度

4、情緒判斷

基於分析的結果，判斷人臉的情緒狀態。針對張嘴和眉毛的狀態進行分類，例如驚訝、開心、生氣等，如下圖 8。

```
if round(mouth_height >= 0.03):
    if eye_hight >= 0.056:
        cv2.putText(im_rd, "amazing", (d.left(), d.bottom() + 20),
cv2.FONT_HERSHEY_SIMPLEX, 1,
                                (0,0,255), 2, 4)
    else:
        cv2.putText(im_rd, "happy", (d.left(), d.bottom() + 20),
cv2.FONT_HERSHEY_SIMPLEX, 1,
                                (0,0,255), 2, 4)
else:
    if self.brow_k <= -0.3:
        cv2.putText(im_rd, "angry", (d.left(), d.bottom() + 20),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,255), 2, 4)
        cv2.putText(im_rd, "nature", (d.left(), d.bottom() + 20),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0,0,255), 2, 4)
```

圖 8：分析人臉情緒狀態之程式碼於 VS Code 撰寫程式之截圖

5、執行效果：如下圖 9、10 所示，可以看見系統成功捕捉到人臉及 68 個特徵點，並成功判斷出當事人之情緒。

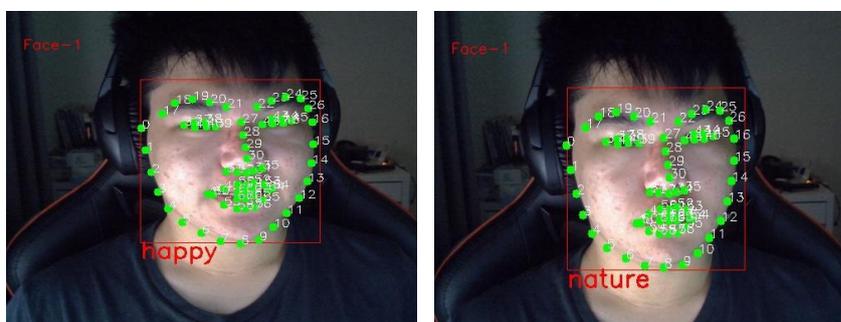


圖 9、10：成功情緒辨識

(四) 口罩下的情緒辨識

我們現階段已完成裸臉的情緒辨識，在能夠破解口罩下的情緒之前，我們必須讓程式偵測判斷人臉是否戴上口罩。同樣利用 dlib 的 68 點模型官方訓練好的特徵預測模型。

1、首先，我們將顏色判定設為 BGR 格式，因 OpenCV 使用 BGR 影像格式。因此，當我們使用 cv2.匯入影像時，它預設以 BGR 格式解釋。接著，使用 dlib 的人臉檢測器 detector 檢測攝像頭畫面中的人臉位置。接著，對於每個檢測到的人臉，程式取得其座標 (x1, y1, x2, y2)，並將人臉區域分為上半部分 (upface) 和下半部分 (downface)。

2、在分割好的區域中，計算了上下兩部分的直方圖 (hist1 和 hist2; cv.calcHist)，然後進行歸一化，以確保直方圖值在 [0, 1] 的範圍內。接下來，使用 OpenCV 比較函式分析這兩個直方圖，得到相似度 near。當 near 小於預定的閾值時，系統判定該人臉佩戴了口罩，並以綠色框和文字顯示 "Wear Mask"; 反之，以紅色框和文字顯示 "No Mask"，如圖 11。

```

# 平移縮放
cv.normalize(hist1, hist1, 0, 1.0, cv.NORM_MINMAX)
cv.normalize(hist2, hist2, 0, 1.0, cv.NORM_MINMAX)

near = cv.compareHist(hist1,hist2,0)

if(near < near_threshold):
    # 畫框框與寫字
    cv.rectangle(frame, (x1, y1), (x2, y2), (0,255, 0), 4,cv.LINE_AA)
    cv.putText(frame,"Wear Mask", (x1, y1), cv.FONT_HERSHEY_DUPLEX,0.7, (0,255, 0), 1, cv.LINE_AA)
else:
    cv.rectangle(frame, (x1, y1), (x2, y2), (0,0,255), 4,cv.LINE_AA)
    cv.putText(frame,"No Mask", (x1, y1), cv.FONT_HERSHEY_DUPLEX,0.7, (0,0,255), 1, cv.LINE_AA)

```

圖 11：判斷人臉是否被口罩遮蔽之程式碼

3、最後，系統使用 cv.imshow () 實時顯示人臉口罩檢測的結果，再與上述情緒辨識之程式整合後就能夠呈現口罩下的情緒辨識，以下為有戴口罩如圖 12 和沒戴口罩如圖 13 的辨識狀況。



圖 12、13：程式執行效果，分別為沒戴口罩 (左，圖 12) 和有戴口罩 (右，圖 13)

五、結論與生活應用

一、結論&改進方向

(一) 發展與實際操作情緒辨識系統：我們成功建立了開發環境，並進行了實際操作以驗證情緒辨識系統的可行性。

(二) 口罩下情緒辨識的整合：透過將口罩因素融入情緒辨識系統，我們獨特地關注在戴口罩情況下的情緒表達。這為實際生活中口罩普及的情境下情緒辨識系統的發展提供了新的可能。

(三) 情緒辨識的原則與應用：透過研究情緒辨識的原理，我們深入探討了不同情緒辨識的種類、功能與應用。起先理解情緒辨識在各個領域的潛在價值，最後我們比對市面上現有的情緒辨識系統，分析優劣與限制，在日後做研究或應用方面提供選擇上的參考。

(四) 成功率提升：1、整合多模式資訊：結合臉部表情、眼神、聲音等多種情緒表達方式，提高辨識成功率。2、光線及距離問題解決：環境光源調整：優化系統以應對不同光線環境，提高辨識效果。3、距離適應性：開發能夠適應不同距離的辨識系統，使其在近距離和遠距離下均能有效辨識情緒。

(五) 系統完善化：使用者友好介面，設計易於操作且直觀的使用者界面，提高系統的實用性和普及度;持續更新和改進，定期更新演算法和系統，以應對不斷變化的使用場景和需求，確保系統的準確性和效能。

二、生活應用

(一) 社交娛樂：配合視訊電話、打卡上傳照片等場合，情緒辨識技術可提升溝通的準確性與

效率，使得遠距溝通更加貼近面對面的真實感，讓戴口罩的人也能在視訊或拍攝限時動態時使用，出現對應的表情貼圖或不同情緒的音樂，增添娛樂性。

(二) 社會安全：台灣近年來詐騙案件層出不窮，受害者往往在前往提款機匯款時才發現受騙，此時若能提前辨識到其焦躁不安的情緒，即可為警方或行員提供寶貴的先兆，以避免犯罪的發生。

(三) 企業服務：即使戴上口罩，能夠根據顧客當下的情緒來調整服務內容，將使得服務更具有個性化與人性化，提升顧客滿意度。

參考資料

王天慶 (2019)。Python 人臉識別從入門到工程實踐。機械工業出版社

陳昭明 (2022)。深度學習：最強入門邁向 AI 專題實戰。深智數位股份有限公司

扶松柏 (2022)。圖像識別技術與實戰

(OpenCV+dlib+Keras+Sklearn+TensorFlow)。清華大學出版社

陳芝瑄、史晨妤 (2022)。生成人臉口罩去除圖像。臺北市立第一女子高級中學：中華民國第 62 屆中小學科學展覽會。(<https://twsf.ntsec.gov.tw/activity/race-1/>

[62/pdf/NPHSF2022-052506.pdf?0.6897529464636916](https://twsf.ntsec.gov.tw/activity/race-1/62/pdf/NPHSF2022-052506.pdf?0.6897529464636916))

李大維 (2019)。智能全縣辨識軟體的開發。朝陽科技大學資訊工程系：碩士論文。

(<http://ir.lib.cyut.edu.tw:8080/bitstream/310901800/37086/1/107CYUT0392010-003.pdf>)

CH.Tseng (2021 年 5 月 10 日)。應用 TripletLoss 於口罩臉孔的辨識。

(https://chtseng.wordpress.com/2021/05/10/tripletloss_training_on_mask_faces/)