

【2024 全國科學探究競賽-這樣教我就懂】

國中組 成果報告表單

題目名稱: 自製瑜珈健身鏡app

一、摘要

本研究主題主要是呼應聯合國永續發展目標(SDGs)第3項「健康與福祉」,以「瑜珈健身鏡」為主題,同時學習健康的生活方式瑜珈伸展與肌耐力的訓練等等,攜手邁向健康、永續的生活。本研究旨在探索自製瑜珈鏡的應用程式,以便協助想學習瑜珈動作的人能透過此應用程式來了解各項瑜珈動作的姿勢正確性。

研究目的,首先是尋找適合初學者的瑜珈動作,並建立相對應的動作資料集;其次,利用瑜珈動作資料集來訓練瑜珈動作AI姿勢辨識模型,以準確地辨識使用者的動作;最後,設計並建置一個瑜珈健身鏡應用程式,通過語音反饋等方式提供使用者練習瑜珈的指導。

研究工具是使用Mediapipe工具進行姿勢辨識、Python和Dart等程式語言進行應用程式的開發。在動作辨識方面,乃是請有多年教學經驗的瑜珈老師討論後進行瑜珈動作的挑選、資料集建立、模型訓練和測試等步驟,以驗證模型的準確性和可靠性。

二、探究題目與動機

上了國中之後,課業壓力明顯地愈來愈大,詢問老師是透過什麼方式來舒壓,老師分享她自身是使用學習瑜珈動作來達成舒壓與放鬆的目的,但是在沒有上瑜珈課時卻不知道自己的瑜珈動作的正確性與否,於是本研究乃是透過瑜珈紓壓。但是自己做太枯燥,沒有老師也不知道該怎麼做,便想要做出一個會鼓勵並輔助自己做瑜珈的AI。

三、探究目的與假設

- (一)尋找適合初學者做的瑜珈動作
- (二)建立瑜珈動作資料集
- (三)訓練瑜珈動作AI姿勢辨識模型
- (四)測試瑜珈動作AI姿勢辨識模型
- (五)建置語言模組以達成動作之反饋
- (六)設計瑜珈健身鏡appUI介面
- (七)建置瑜珈健身鏡介面

四、探究方法與驗證步驟

- (一)研究設備及器材:
 - 1.硬體類:電腦、手機
 - 2.軟體類:Mediapipe(姿勢辨識)、python(svc)、Figma(寫手機軟體)
- (二)研究架構:

瑜珈健身鏡

尋找適合
初學者做的
瑜珈動作

建立瑜珈動
作資料集

訓練瑜珈動
作AI姿勢辨
識模型

測試瑜珈動
作AI姿勢辨
識模型

建置語言模
組以達成動
作之反饋

建置瑜珈健
身鏡app UI
介面

建置瑜珈健
身鏡app

上網搜尋

請瑜珈老師協
助拍攝標準動
作

python

和瑜珈老師一
同以實作來測
試模型

使用軟體為
其配音

Figma

Figma

(三)動作辨識與影響辨識的探究

- 1.使用工具:MediaPipe
- 2.姿勢辨識的實作過程:

(1)上網搜尋適合初學者做的瑜珈動作，最終我們挑選出了8個動作：
船式、金字塔式、下犬式、四足式.....等



(2)建立資料集:錄製瑜珈動作。



(3) 模型參數調整

(4) 訓練模型：匯入訓練資料來訓練模型。

```

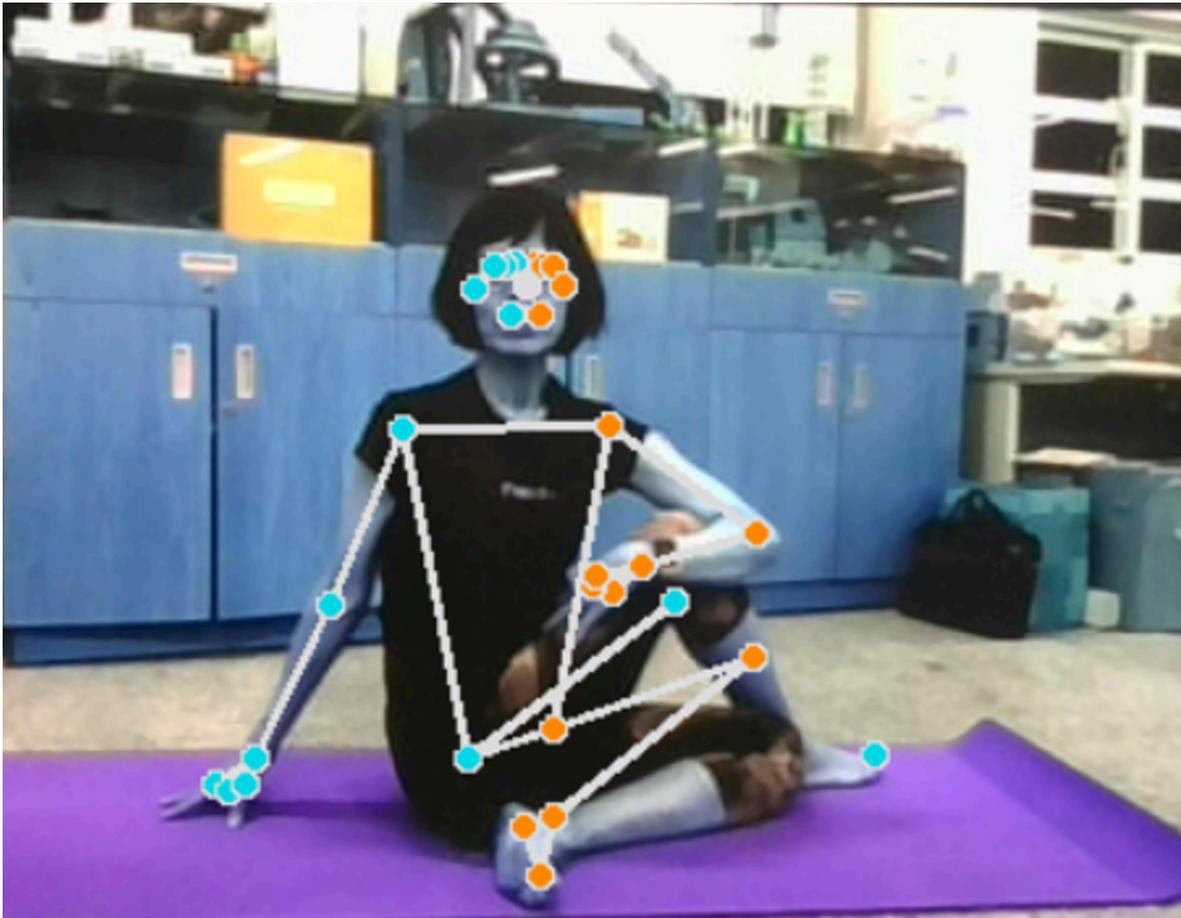
music_box
generateDataset.ipynb  model.joblib  modelTraining.ipynb  testing.ipynb
model > modelTraining.ipynb > X = df.iloc[:, :99].to_numpy()
+ 程式碼 + Markdown | ▶ 全部執行 ⌂ 重新啟動 🗑 清除所有輸出 📄 變數 📖 大綱 ...
base (Python 3.11.7)
1456  0.541726  0.359119  -0.028476  0.550059  0.336432  -0.001659  0.557512  0.335495  -0.001689  0.565037  ...  0.215026  0.550204  -0.559766  0.2396
1457  0.543872  0.362024  -0.024639  0.551237  0.338239  0.002858  0.558622  0.336699  0.002812  0.566016  ...  0.215234  0.551116  -0.537549  0.2415
1458  0.547491  0.364315  -0.025904  0.555451  0.339361  0.002076  0.562659  0.337552  0.002031  0.569488  ...  0.215515  0.539988  -0.526325  0.2444
13156 rows x 100 columns

X = df.iloc[:, :99].to_numpy()
y = df.iloc[:, 99:].to_numpy().flatten()
print(x.shape)
print(y.shape)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.1, random_state=42)
svc = SVC(kernel='linear', probability=True)
svc.fit(X_train, y_train)
y_pred = svc.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Accuracy: {accuracy:.4f}")

(13156, 99)
(13156,)
Accuracy: 0.9916

from joblib import dump
dump(svc, 'modell.joblib')

['modell.joblib']
  
```



(5)測試模型: 導出模型, 讓不同人做不同的瑜珈姿勢, 測試模型能否準確辨識動作。

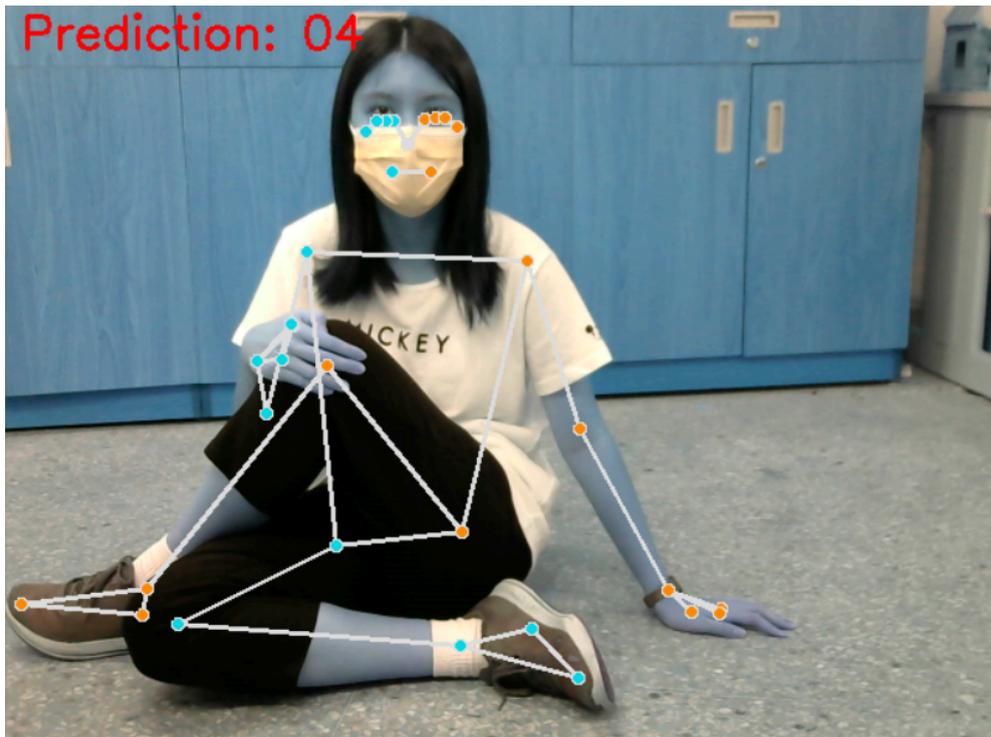
```
music_box
├── arduino
│   └── bluetoothino
│       └── model
│           ├── 00.csv
│           ├── 01.csv
│           ├── 02.csv
│           ├── 03.csv
│           ├── 04.csv
│           ├── 05.csv
│           ├── 06.csv
│           ├── 07.csv
│           ├── 08.csv
│           ├── 09.csv
│           ├── generateDataset.ipynb
│           ├── handsUp.csv
│           ├── headLeft.csv
│           ├── headRight.csv
│           ├── modelJoblib
│           ├── modelJoblib
│           ├── modelTraining.ipynb
│           └── testing.ipynb
├── python
│   ├── bluetooth_recieve.py
│   ├── bluetooth_transmit.py
│   ├── requirements.txt
│   └── README.md
├── 大綱
└── 時間表

generateDataset.ipynb
model
testing.ipynb
with mp_pose.Pose(
    landmark_drawing_spec=mp_drawing_styles.get_default_pose_landmarks_style())

# Display the prediction on the frame
if prediction == 0:
    cv2.putText(img, 'Prediction: 00', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction == 1):
    cv2.putText(img, 'Prediction: 01', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==2):
    cv2.putText(img, 'Prediction: 02', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==3):
    cv2.putText(img, 'Prediction: 03', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==4):
    cv2.putText(img, 'Prediction: 04', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==5):
    cv2.putText(img, 'Prediction: 05', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==6):
    cv2.putText(img, 'Prediction: 06', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==7):
    cv2.putText(img, 'Prediction: 07', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==8):
    cv2.putText(img, 'Prediction: 08', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
elif(prediction==9):
    cv2.putText(img, 'Prediction: 09', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
# Display the resulting frame
# prob = model.predict_proba([features])[0]
# cv2.putText(img, f'confidence {prob}', (10, 30), cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 0, 255), 2)
cv2.imshow('MediaPipe Pose', img)

# Break the loop with the 'q' key
if cv2.waitKey(1) & 0xFF == ord('q'):
    break

cap.release()
cv2.destroyAllWindows()
```



(6)製作App:使用Figma設計App版面並撰寫程式製作成App。



五、結論與生活應用

結論：

本研究旨在探索一種利用瑜珈來紓解學生壓力的方法，並提出了一個創新的解決方案：設計一個能夠鼓勵並輔助初學者進行瑜珈練習的人工智能(AI)系統，即自製瑜珈健身鏡app。透過尋找適合初學者的瑜珈動作、建立相應的動作資料集、訓練瑜珈動作AI姿勢辨識模型以及建置語音反饋等功能，該應用程式有望成為一個有效的學習工具，幫助學生放鬆身心，紓解壓力。

生活應用：

放鬆身心：自製瑜珈健身鏡app為學生提供了一個方便的放鬆途徑。透過瑜珈練習，使用者可以舒緩壓力，放鬆身心，提升心理健康。

室內運動選擇：在現代社會，由於各種因素，室外運動可能受到限制。自製瑜珈健身鏡app提供了一個室內運動的選擇，讓使用者在家中或任何地方都能夠輕鬆進行瑜珈練習。

個性化練習：通過模型訓練和測試，該應用程式可以根據使用者的需求和水平提供個性化的練習建議，使每個人都能夠找到適合自己的瑜珈練習方式。

總的來說，自製瑜珈健身鏡app將人工智能技術與瑜珈練習相結合，為學生提供了一個便捷、有效的放鬆方式，有助於改善他們的身心健康，應用價值和潛力巨大。



參考資料

使用機器學習解決問題的五步驟：模型訓練(2024)。取自：

<https://datasciocean.tech/machine-learning-basic-concept/machine-learning-model-training/>

Pose landmark detection guide for Python(2024)。取自：

https://developers.google.com/mediapipe/solutions/vision/pose_landmarker/python#live-stream

瑜珈初學者必學！6個基礎瑜珈姿勢、優點一次統整！（2022）。取自：

<https://www.lotusfitness.com.tw/blog/posts/yoga>

Figma

https://www.figma.com/login?is_not_gen_0=true