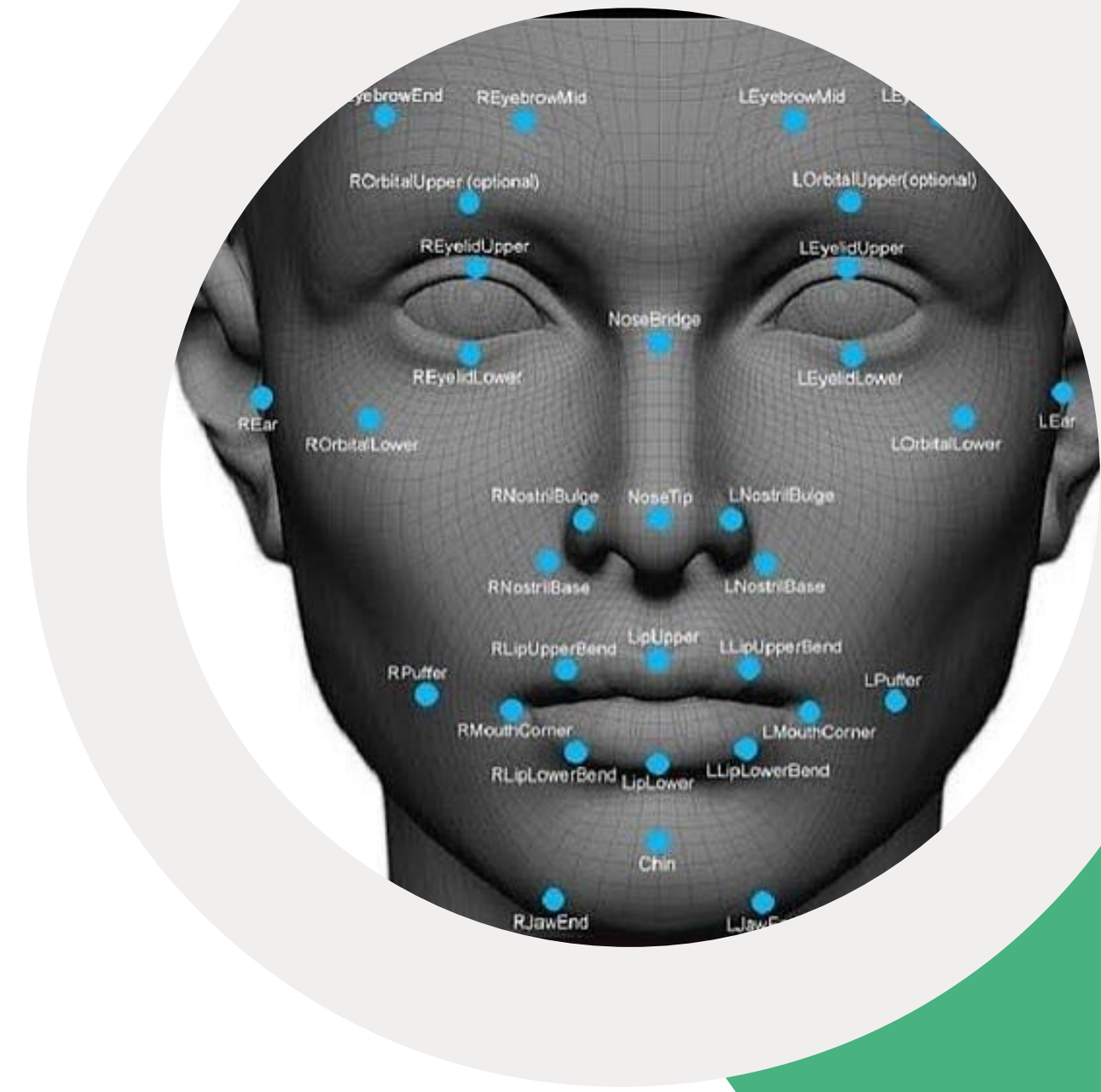


工科海工專論

B10505050 □ □ □ □ □ □



目錄

- 主題發想

- □ □ □ □

- □ □ □ □ □

- LBPH □ □

- □ □ □ □

- demo



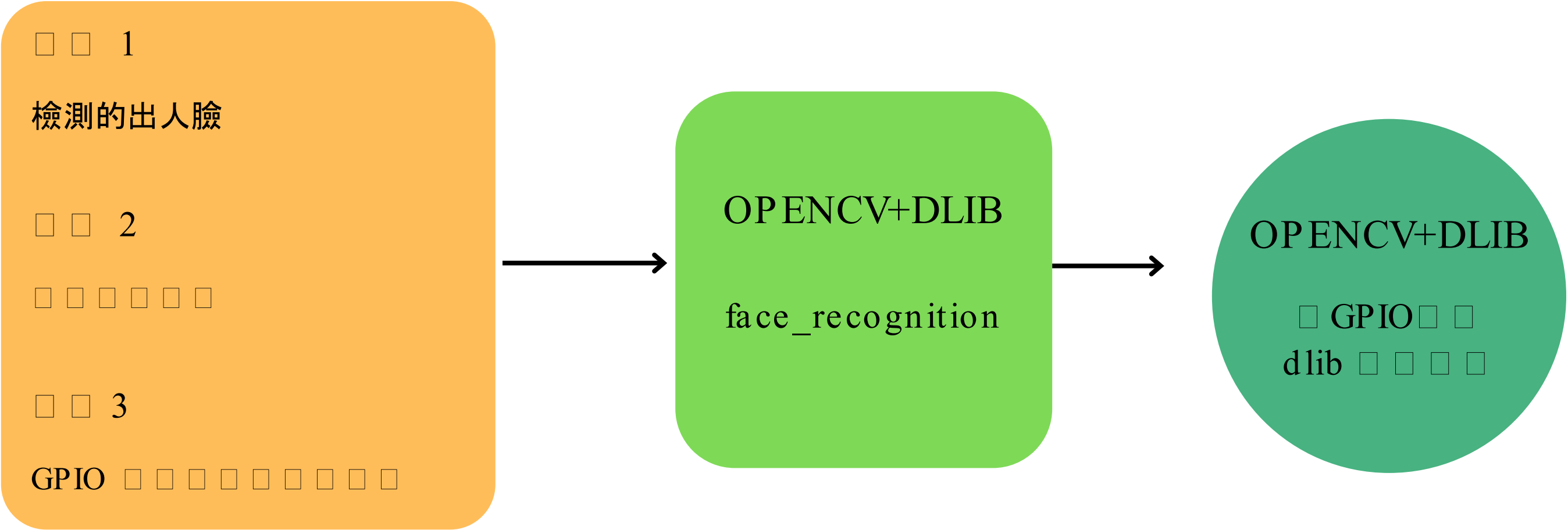
□ □ □ □ □ □

主題發想

□ □ □ □ □ □



最終方案-門禁系統



替代方案

LBP-H

□ □ □ □

□ □ train_model.py □ □ □ □ □ □
□ □ □ □ □ □ □ □ □ □ LBP-H □ □

問題 : □ □ □ □ □

HARRCASCASCADE

□ □ □ □

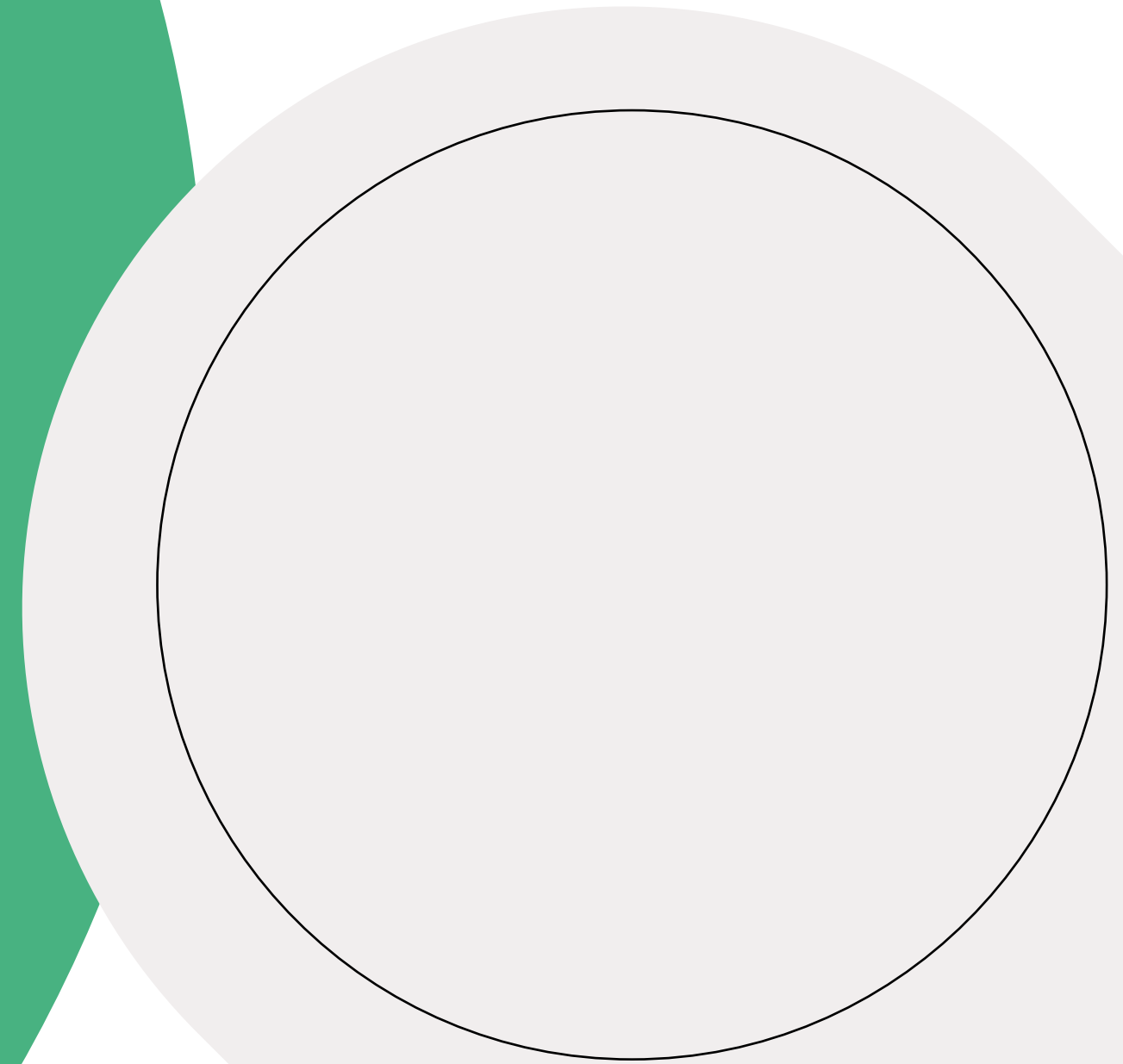
□ □ app2.py □ □ □ □ Flask □ □ □ □
□ □ □ □ □ □ □ □ □ □ □ □ □ □ □

□ □ □ □ □ □ □ □ □ □ □ □ □ □

□ □ : □ □ □ □ □ □

專案架構

```
— data/
  |   └─ known_faces/
  |       └─ Ko/
  |           └─ training_image.jpeg
  └─ models/
      └─ trained_face_recognition_model.ym
  └─ label_dict.json
  └─ train_model.py
  └─ app2.py
```



部分程式碼

TRAIN_MODEL.PY

```
def train_model(data_path, model_save_path, label_dict_path):  
    #引入LBPH  
    face_recognizer = cv2.face.LBPHFaceRecognizer_create()
```

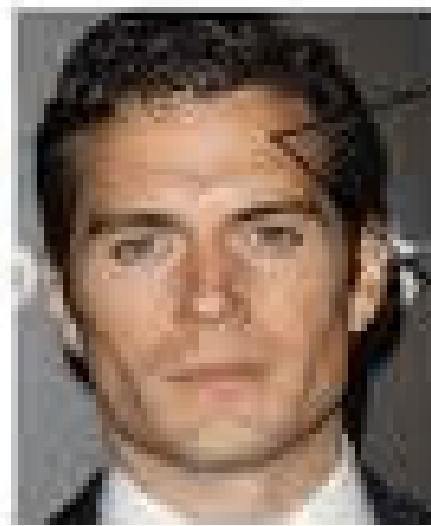
```
#train人臉識別模型  
face_recognizer.train(faces, np.array(labels))  
#保存模型  
face_recognizer.save(model_save_path)  
#存取標籤  
with open(label_dict_path, 'w', encoding='utf-8') as f:  
    json.dump(label_dict, f, ensure_ascii=False)  
return label_dict
```

APP2.PY

```
# 進行人臉檢測  
gray = cv2.cvtColor(image_array, cv2.COLOR_BGR2GRAY)  
gray = cv2.equalizeHist(gray) # 增强图像对比度  
faces = face_cascade.detectMultiScale(  
    gray,  
    scaleFactor=1.1, # scaleFactor(越大越快、但會有疏失)就是縮放比例  
    minNeighbors=5, # minNeighbors 檢測次數  
    minSize=(50, 50) # 檢測尺寸  
)
```

```
for (x, y, w, h) in faces:  
    face = gray[y:y+h, x:x+w]  
    label, confidence = face_recognizer.predict(face)  
    if confidence < 80: #可信度，很像找出來的匹配率標準，越小越像，所以小於此值才會顯示  
        name = label_dict.get(str(label), "Unknown")  
        text = f"{name}: {confidence:.2f}"  
        color = (0, 255, 0) # 設定方框為綠色  
    else:  
        text = "Unknown"  
        color = (0, 0, 255) # 設定方框為紅色  
  
    cv2.rectangle(image_array, (x, y), (x+w, y+h), color, 2)  
    cv2.putText(image_array, text, (x, y-10), cv2.FONT_HERSHEY_SIMPLEX, 0.9, color, 2)
```

LBPH簡介



3*3 pixels

200	50	50
50	90	100
160	70	210

Threshold
90

1	0	0
0		1
1	0	1

Binary
10001101

150	90	80
30	141	

Decimal
141

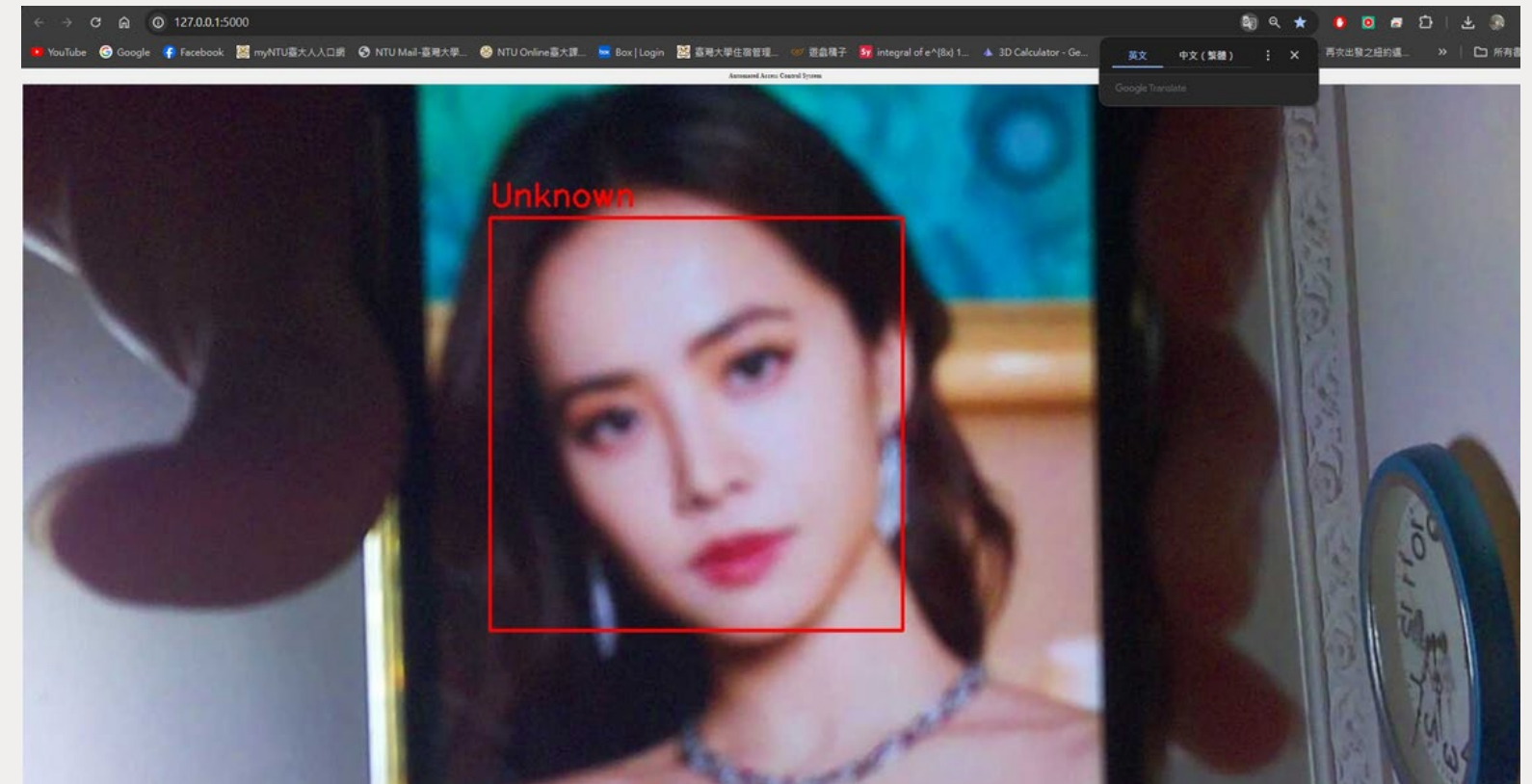
S
(□ □)

W
(□ □)

- 算法簡單, 易於理解
- □ □ □ □ □ □ □ □ □ □ □ □ □ □

- □ □ □ □ □
- □ □ □ □ □ □ □ □ □ □ □ □ □ □ □ □

實作截圖





demo

Thanks
for
Listening