

Nhận diện khuôn mặt với FaceNet

Notebook này trình bày hệ thống nhận diện khuôn mặt thời gian thực sử dụng mô hình FaceNet. Hệ thống sử dụng webcam để quay video, phát hiện khuôn mặt và so sánh với hình ảnh đã biết để xác định mức độ tương đồng.

```
In [ ]: import cv2
import numpy as np
from mtcnn import MTCNN
from keras_facenet import FaceNet
from scipy.spatial.distance import cosine
```

Nhập các thư viện cần thiết

Các thư viện sau được nhập:

- `cv2` : Xử lý hình ảnh và video.
- `numpy` : Các phép toán số học.
- `MTCNN` : Phát hiện khuôn mặt.
- `FaceNet` : Tạo embedding cho khuôn mặt.
- `cosine` từ `scipy.spatial.distance` : Tính toán độ tương đồng giữa các embedding.

```
In [ ]: detector = MTCNN()
embedder = FaceNet()

# Kiểm tra xem webcam có hoạt động không
cap = cv2.VideoCapture(0)
if not cap.isOpened():
    print("Không thể mở webcam. Vui lòng kiểm tra kết nối hoặc quyền truy cập.")
    cap.release()
    exit()
```

Khởi tạo bộ phát hiện khuôn mặt và tạo embedding

- `MTCNN` được sử dụng để phát hiện khuôn mặt trong hình ảnh.
- `FaceNet` được sử dụng để tạo embedding cho các khuôn mặt được phát hiện.

Ngoài ra, webcam được khởi tạo để quay video thời gian thực.

```
In [3]: def get_embedding(face_img):
    face_img = cv2.resize(face_img, (160, 160))
    face_img = face_img.astype('float32')
    face_img = np.expand_dims(face_img, axis=0)
```

```
embedding = embedder.embeddings(face_img)
return embedding[0]
```

Định nghĩa hàm tạo embedding

Hàm `get_embedding` xử lý trước hình ảnh khuôn mặt và tạo embedding của nó bằng mô hình FaceNet. Embedding này là biểu diễn số của khuôn mặt, có thể được sử dụng để so sánh độ tương đồng.

```
In [4]: known_img = cv2.imread("anh-the.png")
known_img = cv2.cvtColor(known_img, cv2.COLOR_BGR2RGB)

faces = detector.detect_faces(known_img)
x, y, w, h = faces[0]['box']
known_face = known_img[y:y+h, x:x+w]

known_embedding = get_embedding(known_face)
```

1/1 —————— 4s 4s/step

Tải và xử lý hình ảnh đã biết

- Hình ảnh đã biết được tải và chuyển đổi sang định dạng RGB.
- Khuôn mặt trong hình ảnh được phát hiện, cắt và truyền vào hàm `get_embedding` để tạo embedding.

Embedding này sẽ được sử dụng làm tham chiếu để so sánh độ tương đồng.

```
In [5]: cap = cv2.VideoCapture(0)

THRESHOLD = 0.7 # ngưỡng similarity

while True:
    ret, frame = cap.read()
    if not ret:
        break

    rgb_frame = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

    try:
        faces = detector.detect_faces(rgb_frame)
    except Exception as e:
        print("Lỗi detect, bỏ qua frame này...")
        continue

    for face in faces:
        x, y, w, h = face['box']
        face_img = rgb_frame[y:y+h, x:x+w]

        try:
            emb = get_embedding(face_img)
```

```
similarity = 1 - cosine(know_embedding, emb)

if similarity > THRESHOLD:
    label = "Matched"
    color = (0, 255, 0)
else:
    label = "Unknown"
    color = (0, 0, 255)

cv2.rectangle(frame, (x, y), (x+w, y+h), color, 2)
cv2.putText(
    frame,
    f"{label} {similarity:.2f}",
    (x, y-10),
    cv2.FONT_HERSHEY_SIMPLEX,
    0.8,
    color,
    2
)

except Exception as e:
    pass

cv2.imshow("Face Recognition - FaceNet", frame)
# Kiểm tra nếu cửa sổ bị đóng
if cv2.getWindowProperty("Face Recognition - FaceNet", cv2.WND_PROP_VISIBLE) <
    print("Cửa sổ bị đóng, thoát chương trình...")
    break

if cv2.waitKey(1) & 0xFF == 27: # ESC
    break

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

1/1 0s 128ms/step
1/1 0s 178ms/step
1/1 0s 124ms/step
1/1 0s 142ms/step
1/1 0s 146ms/step
1/1 0s 121ms/step
1/1 0s 124ms/step
1/1 0s 123ms/step
1/1 0s 121ms/step
1/1 0s 177ms/step
1/1 0s 125ms/step
1/1 0s 172ms/step
1/1 0s 160ms/step
1/1 0s 152ms/step
1/1 0s 116ms/step
1/1 0s 190ms/step
1/1 0s 111ms/step
1/1 0s 287ms/step
1/1 0s 152ms/step
1/1 0s 110ms/step
1/1 0s 132ms/step
1/1 0s 164ms/step
1/1 0s 182ms/step
1/1 0s 141ms/step
1/1 0s 112ms/step
1/1 0s 170ms/step
1/1 0s 137ms/step
1/1 0s 129ms/step
1/1 0s 159ms/step
1/1 0s 163ms/step
1/1 0s 256ms/step
1/1 0s 182ms/step
1/1 0s 110ms/step
1/1 0s 145ms/step
1/1 0s 113ms/step
1/1 0s 127ms/step
1/1 0s 100ms/step
1/1 0s 108ms/step
1/1 0s 101ms/step
1/1 0s 111ms/step
1/1 0s 114ms/step
1/1 0s 102ms/step
1/1 0s 117ms/step
1/1 0s 105ms/step
1/1 0s 120ms/step
1/1 0s 109ms/step
1/1 0s 126ms/step
1/1 0s 144ms/step
1/1 0s 124ms/step
1/1 0s 100ms/step
1/1 0s 121ms/step
1/1 0s 94ms/step
1/1 0s 108ms/step
1/1 0s 112ms/step
1/1 0s 109ms/step
1/1 0s 137ms/step

1/1 0s 116ms/step
1/1 0s 127ms/step
1/1 0s 108ms/step
1/1 0s 113ms/step
1/1 0s 160ms/step
1/1 0s 98ms/step
1/1 0s 109ms/step
1/1 0s 110ms/step
1/1 0s 120ms/step
1/1 0s 112ms/step
1/1 0s 120ms/step
1/1 0s 134ms/step
1/1 0s 100ms/step
1/1 0s 107ms/step
1/1 0s 109ms/step
1/1 0s 115ms/step
1/1 0s 125ms/step
1/1 0s 113ms/step
1/1 0s 101ms/step
1/1 0s 103ms/step
1/1 0s 99ms/step
1/1 0s 106ms/step
1/1 0s 121ms/step
1/1 0s 102ms/step
1/1 0s 110ms/step
1/1 0s 104ms/step
1/1 0s 101ms/step
1/1 0s 102ms/step
1/1 0s 116ms/step
1/1 0s 115ms/step
1/1 0s 137ms/step
1/1 0s 96ms/step
1/1 0s 101ms/step
1/1 0s 113ms/step
1/1 0s 111ms/step
1/1 0s 119ms/step
1/1 0s 105ms/step
1/1 0s 101ms/step
1/1 0s 108ms/step
1/1 0s 101ms/step
1/1 0s 109ms/step
1/1 0s 114ms/step
1/1 0s 128ms/step
1/1 0s 107ms/step
1/1 0s 108ms/step
1/1 0s 102ms/step
1/1 0s 120ms/step
1/1 0s 139ms/step
1/1 0s 162ms/step
1/1 0s 100ms/step
1/1 0s 104ms/step
1/1 0s 112ms/step
1/1 0s 98ms/step
1/1 0s 99ms/step
1/1 0s 98ms/step
1/1 0s 95ms/step

1/1 ━━━━━━ 0s 99ms/step
1/1 ━━━━ 0s 105ms/step
1/1 ━━━━ 0s 98ms/step

Vòng lặp nhận diện khuôn mặt thời gian thực

- Quay video từ webcam.
- Phát hiện khuôn mặt trong từng khung hình và tạo embedding cho chúng.
- So sánh embedding với embedding đã biết để tính toán độ tương đồng.
- Hiển thị khung và nhãn ("Matched" hoặc "Unknown") xung quanh các khuôn mặt được phát hiện dựa trên ngưỡng độ tương đồng.

Nhấn `ESC` để thoát vòng lặp.