

人臉辨識

一、OpenCV 人臉分類器

01. 影像測試

A. 匯入模塊函數

【01】

```
import cv2
import dlib
import matplotlib.pyplot as plt
import time
```

B. 使用 OpenCV 訓練完成的人臉辨識 Haar 分類模型：

- 回傳的人臉位置為左上角的 x, y 與寬高 w, h
- 回傳的格式為陣列 (Array) 的型式

【02】

```
img = cv2.imread("Face_test.jpg")
img = cv2.resize(img, None, fx=1.5, fy=1.5)

cv_detector = cv2.CascadeClassifier('haarcascades/haarcascade_frontalface_alt2.xml')

start = time.time()
cv_faces = cv_detector.detectMultiScale(img, scaleFactor=1.2, minNeighbors=3, minSize=(15, 15), flags = cv2.CASCADE_SCALE_IMAGE)
end = time.time()
print("time:%0.3f" %(end - start))

for i in range(len(cv_faces)):
    x = cv_faces[i][0]
    y = cv_faces[i][1]
    w = cv_faces[i][2]
    h = cv_faces[i][3]

    cv2.rectangle(img, (x, y), (x+w, y+h), (0, 0, 255), 4)

plt.figure(figsize=(15,10))
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
plt.show()
```

1. time.time(): 返回當前時間的時間點



02. 攝影機測試

A. 使用 OpenCV 訓練完成的人臉辨識 Haar 分類模型：

【03】	<pre>VIDEO_IN = cv2.VideoCapture(0) while True: hasFrame, img = VIDEO_IN.read() img = cv2.resize(img, None, fx=0.6, fy=0.6) cv_faces = cv_detector.detectMultiScale(img, scaleFactor=1.2, minNeighbors=3, minSize=(15, 15), flags = cv2.CASCADE_SCALE_IMAGE) for i in range(len(cv_faces)): x = cv_faces[i][0] y = cv_faces[i][1] w = cv_faces[i][2] h = cv_faces[i][3] cv2.rectangle(img, (x, y), (x+w, y+h), (0, 0, 255), 4) cv2.imshow("Frame", img) if cv2.waitKey(1) & 0xFF == ord('q'): break VIDEO_IN.release() cv2.destroyAllWindows()</pre>
------	--

二、Dlib 人臉分類器

01. 影像測試

A. 使用 Dlib 訓練完成的人臉辨識分類模型：

- 回傳的人臉位置為左上角的 x1, y1 與右下角的 x2, y2
- 回傳的格式為 dlib.rectangle 的型式，若要提取其值則是使用 left、top、right 與 bottom

【04】	<pre>img = cv2.imread("Face_test.jpg") img = cv2.resize(img, None, fx=1.5, fy=1.5) dlib_detector = dlib.get_frontal_face_detector() start = time.time() dlib_faces = dlib_detector(img) end = time.time() print("time:%0.3f" %(end - start))</pre>
------	---

```

for i in range(len(dlib_faces)):
    x1 = dlib_faces[i].left()
    y1 = dlib_faces[i].top()
    x2 = dlib_faces[i].right()
    y2 = dlib_faces[i].bottom()

    cv2.rectangle(img, (x1, y1), (x2, y2), (0, 0, 255), 4)

plt.figure(figsize=(15,10))
plt.imshow(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
plt.show()

```

02. 攝影機測試

A. 使用 Dlib 訓練完成的人臉辨識分類模型：

【05】

```

VIDEO_IN = cv2.VideoCapture(0)
while True:
    hasFrame, img = VIDEO_IN.read()
    img = cv2.resize(img, None, fx=0.6, fy=0.6)
    dlib_faces = dlib_detector(img)

    for i in range(len(dlib_faces)):
        x1 = dlib_faces[i].left()
        y1 = dlib_faces[i].top()
        x2 = dlib_faces[i].right()
        y2 = dlib_faces[i].bottom()

        cv2.rectangle(img, (x1, y1), (x2, y2), (0, 0, 255), 4)
        cv2.imshow("Frame", img)

        if cv2.waitKey(1) & 0xFF == ord('q'):
            break

VIDEO_IN.release()
cv2.destroyAllWindows()

```