

WORKSHOP RECONNAISSANCE FACIALE



Connaissez vous la différence entre détection et reconnaissance faciale ?

Nous allons durant ce workshop apprendre à manier des deux concepts simplement grâce à l'utilisation d'une librairie en Python.

PRÉREQUIS

- Python3
- pip3
- Librairie face recognition

```
rafou@rafou-thinkpad-t470s:~$ pip3 install face_recognition
```

- Téléchargement des images

https://github.com/RaphaelDk/Facial_Recognition

PROGRAMMATION

I/ Lecture du flux vidéo de la webcam

```
1  import cv2~
2  ~
3  #Initialisation de la webcam~
4  video_capture = cv2.VideoCapture(0)~
5  ~
6  while True:~
7      ...#Obtention de l'image~
8      ...ret, frame = video_capture.read()~
9      ~
10     ...#Affichage~
11     ...cv2.imshow('Video', frame)~
12     ...if cv2.waitKey(1) & 0xFF == ord('q'):~
13         ...break~
14     ~
15 video_capture.release()~
16 cv2.destroyAllWindows()~
```

II/ Détection des visages présents sur l'image

```
7  #Permet de ne traiter qu'une image sur deux~
8  process_this_frame = True~
9  #Stockage de la localisation des visages~
10 face_locations = []~
~
16 ...#Resize de l'image a 1/4 afin d'optimiser les performances~
17 ...small_frame = cv2.resize(frame, (0, 0), fx=0.25, fy=0.25)~
18 ...#Conversion du format BGR (OpenCV) vers RGB (face_recognition)~
19 ...rgb_small_frame = small_frame[:, :, :-1]~
20 ~
21 ...if process_this_frame:~
22     ...#Localiser les visages sur l'image~
23     ...face_locations = face_recognition.face_locations(rgb_small_frame)~
24     ~
25     process_this_frame = not process_this_frame~
26 ~
27     ...for (top, right, bottom, left) in face_locations:~
28         ...#Faire correspondre les coordonnees au resize effectue~
29         ...top *= 4~
30         ...right *= 4~
31         ...bottom *= 4~
32         ...left *= 4~
33     ~
34     ...#Affichage d'un rectangle autour des visages~
35     ...cv2.rectangle(frame, (left, top), (right, bottom), (0, 0, 255), 2)~
```

III/ Apprentissage de la reconnaissance de certains visages

```
12 #Chargement des visages des personnes a reconnaître~
13 raphael_image = face_recognition.load_image_file("raphael.jpg")~
14 scarlett_image = face_recognition.load_image_file("scarlett.jpg")~
15 ~
16 #Apprentissage de la reconnaissance de ces visages~
17 raphael_face_encoding = face_recognition.face_encodings(raphael_image)[0]~
18 scarlett_face_encoding = face_recognition.face_encodings(scarlett_image)[0]~
19 ~
20 known_face_encodings = [~
21     raphael_face_encoding,~
22     scarlett_face_encoding~
23 ]~
24 known_face_names = [~
25     "Raphael",~
26     "Scarlett"~
27 ]~
```

IV/ Reconnaissance des visages sur l'image

```
11 #Reconnaissance des visages~
12 face_encodings = []~
13 face_names = []~

44 ~~~~~#Reconnaissance des visages localises~
45 ~~~~~face_encodings = face_recognition.face_encodings(rgb_small_frame,~
46 ~~~~~face_locations)~
47 ~
48 ~~~~~face_names = []~
49 ~~~~~for face_encoding in face_encodings:~
50 ~~~~~~#Verification des correspondences avec les visages enregistres~
51 ~~~~~~matches = face_recognition.compare_faces(known_face_encodings,~
52 ~~~~~~face_encoding)~
53 ~
54 ~~~~~~#Enregistrement du nom de la personne reconnue~
55 ~~~~~~name = "Unknown"~
56 ~~~~~~if True in matches:~
57 ~~~~~~|~first_match_index = matches.index(True)~
58 ~~~~~~|~name = known_face_names[first_match_index]~
59 ~~~~~~face_names.append(name)~

73 ~~~~~~#Affichage du nom de la personne reconnue~
74 ~~~~~~cv2.rectangle(frame, (left, bottom - 35), (right, bottom),~
75 ~~~~~~(0, 0, 255), cv2.FILLED)~
76 ~~~~~~cv2.putText(frame, name, (left + 6, bottom - 6),~
77 ~~~~~~cv2.FONT_HERSHEY_DUPLEX, 1.0, (255, 255, 255), 1)~
```