Project : Face Detection And Recognition

(Software Module)

The Software module is designed and developed to be applied for Real time mode of detection and recognition of face for example in door bell/door lock access

Camera	→ Signal →	Software Module	→ Signal →	Allow (Access Granted)
				Do Not Allow (Access Not Granted)

Software Architecture of Door Lock

- Create Dataset
- Detect faces and Compute 128-d face embeddings to quantify a face
- Train a Support Vector Machine (SVM) on top of the embeddings
- Recognize faces in images and video streams

Dataset Creation

Dataset is created by capturing the image containing face with different positions with different poses and scales to provide reasonable accuracy. We may even perform this process over multiple days or weeks to gather examples of their face in:

- Different lighting conditions
- Times of day
- Moods and emotional states

Algorithm:

1. Create the folder with a unique id(name).
2. Store the images captured from a different position, angle, poses, different light and resolution conditions to that folder.

Training Phase

We pass the detected face through the embedder CNN. This generates a 128-D vector (vector matrix of image) which describes the face. And Label-Encoder encode our name with the matrix.

Algorithm:

1. Detect the ROI (face) from the image.
2. Create the 128-D vector matrix of that image and encode the name(label encoder).
3. With the use of SVM(Support Vector Machine) the 128-d matrix & the encoded label will be stored in one pickle file.

Execution

Detection Phase

In Detection phase the face will be detected from the live video stream. We'll use the OpenCV's deep learning face detector model to detect where in the image the face ROIs are.
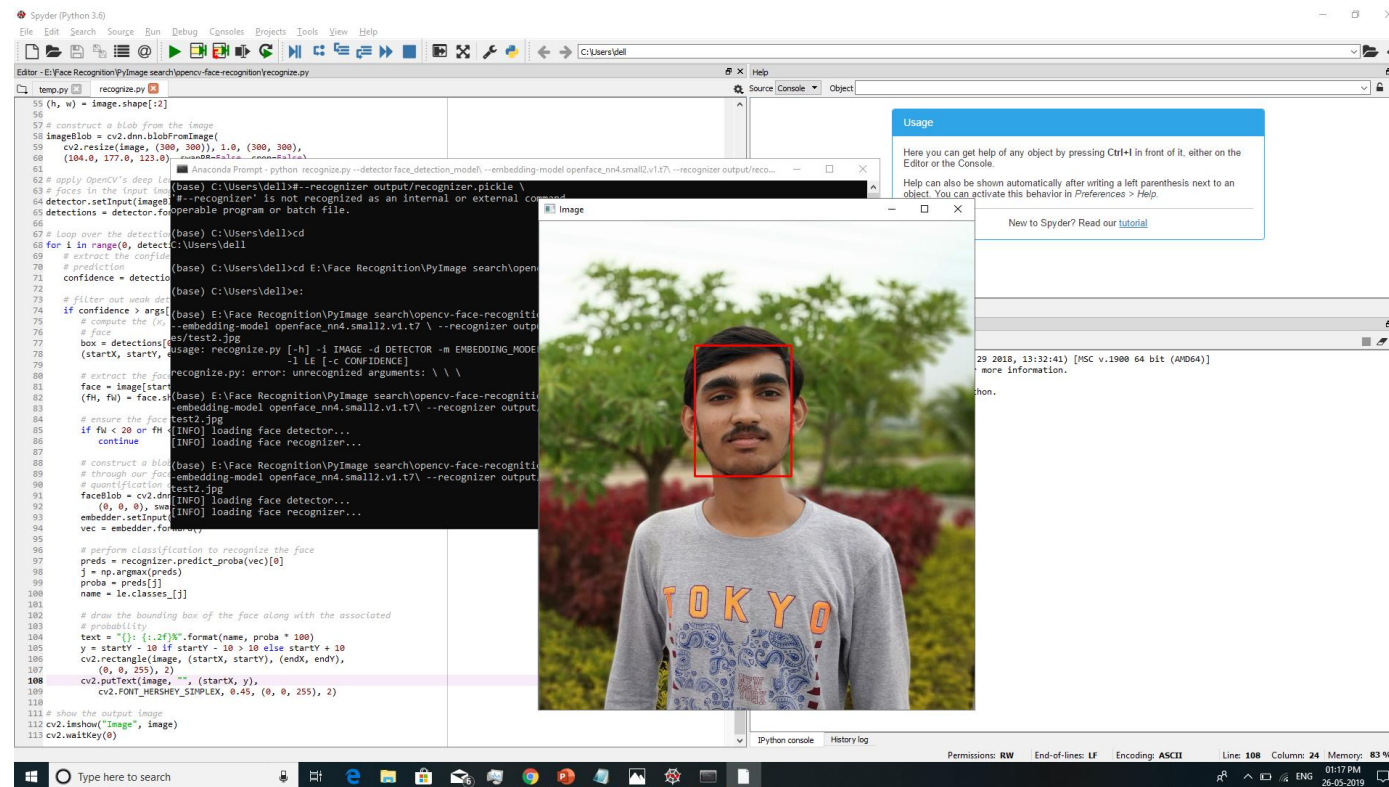

Fig 1: Detect face

Recognition Phase

In recognition phase device scans a human face, it then matches the identified image against the enrolled image that is stored in face recognition database.

Algorithm:

1. Using the OpenCV's deep learning face detector model detect the ROI (face) from the Video stream.
2. Using OpenCV's deep learning face embedding model. We'll use this model to extract the 128-D face embedding from the face ROI and feed the data into the recognizer.
3. From the trained SVM recognizer extract the encoded label.
4. The optional threshold to filter weak face detections (confidence).
   - Confidence level more then 75%
   - Tolerance level 2%
   - Support Vector Machine model
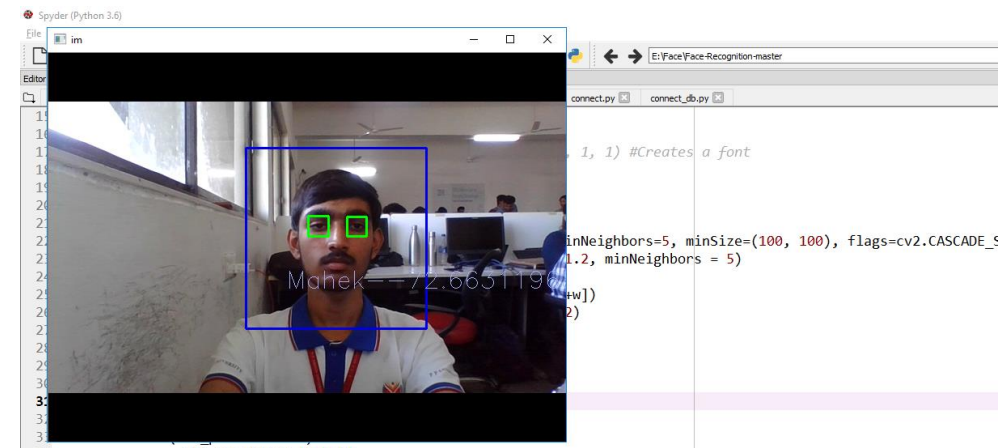
Fig 2: Face Recognition
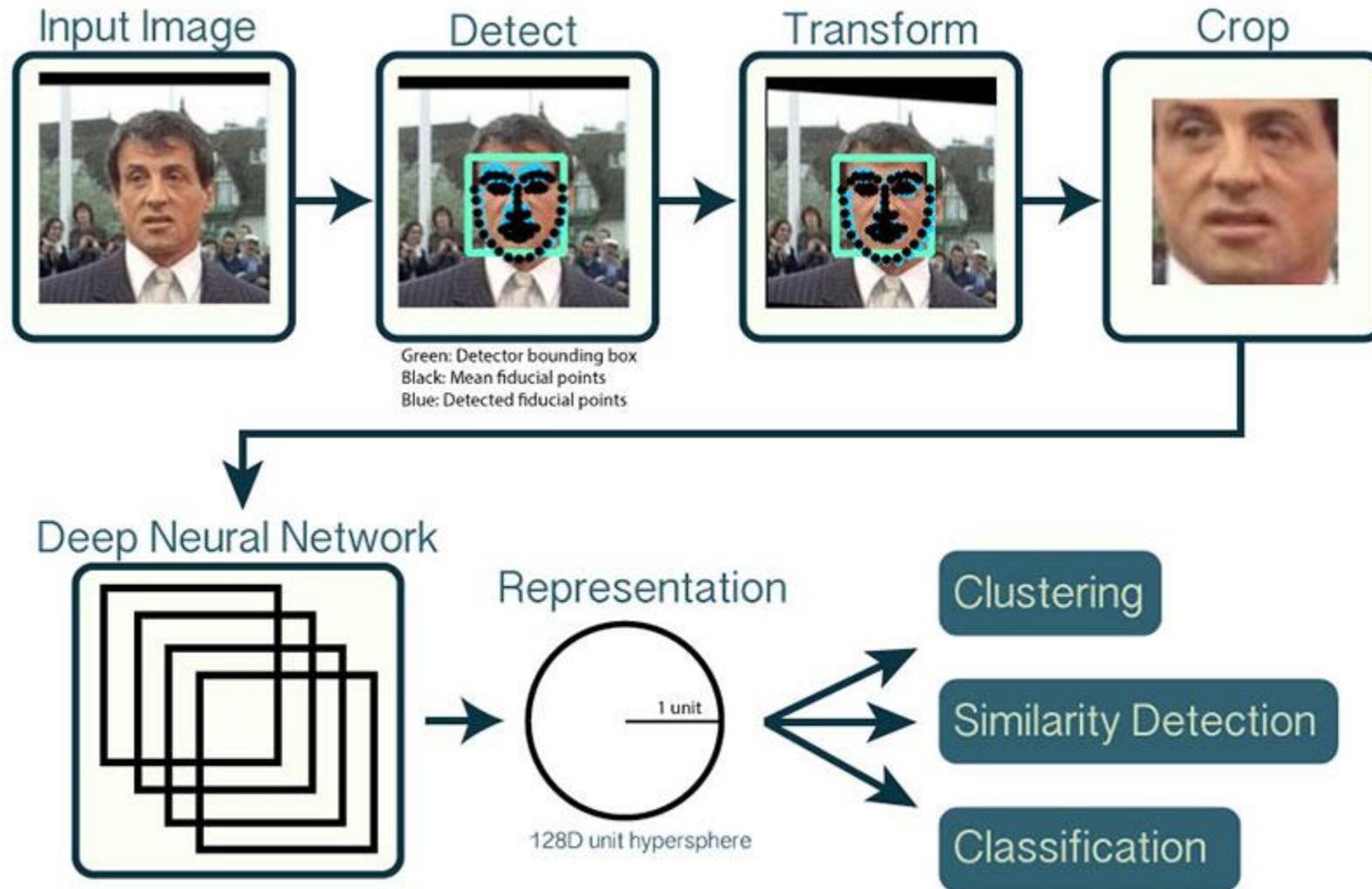
Fig 3: An overview of the OpenCV face recognition pipeline. The key step is a CNN feature extractor that generates 128-d facial embeddings.