

OpenCV Face Recognition

In today's project, you will learn how to perform face recognition using the OpenCV library.

face recognition post relied on two important external libraries:

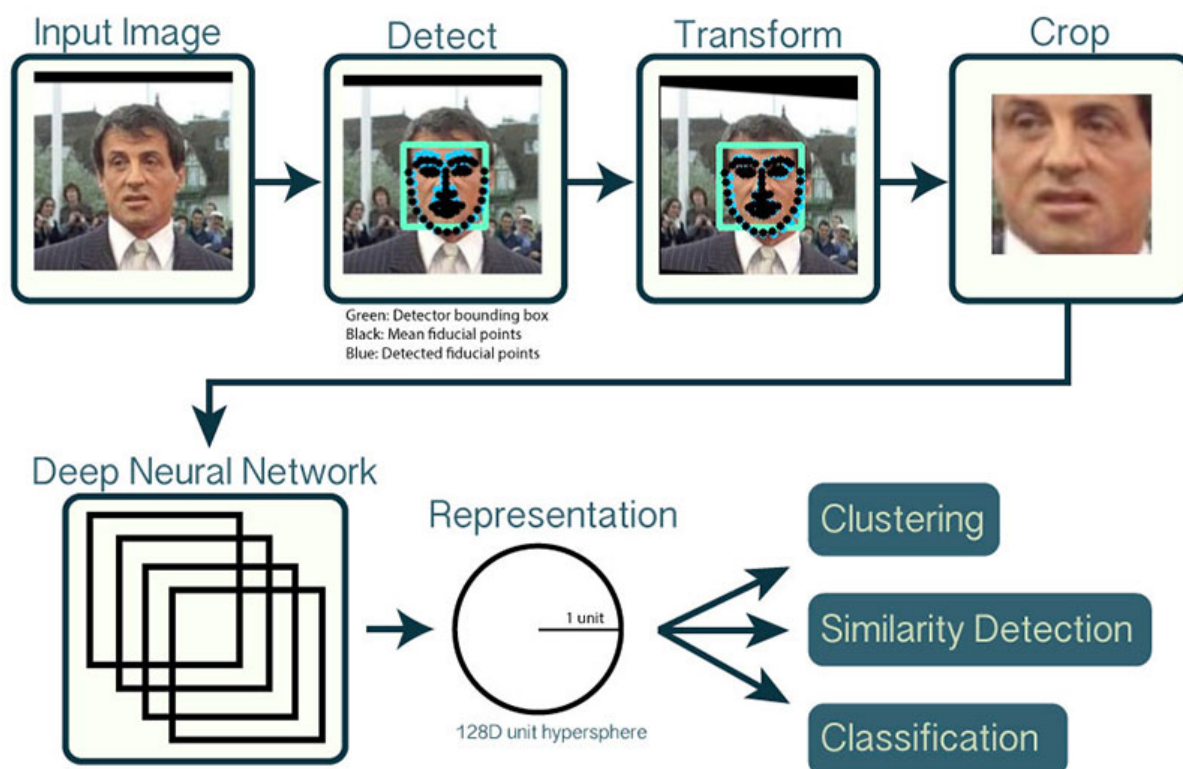
1. [dlib](#) (obviously)
2. [face_recognition](#) (which is an easy to use set of face recognition utilities that wraps around dlib)

In today's project, we'll learn how we can apply deep learning and OpenCV together to:

- Detect faces
- 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
- All of these tasks will be accomplished with OpenCV, enabling us to obtain a "pure" OpenCV face recognition pipeline.

How OpenCV's face recognition works

In order to build our OpenCV face recognition pipeline, we'll be applying deep learning in two key steps:

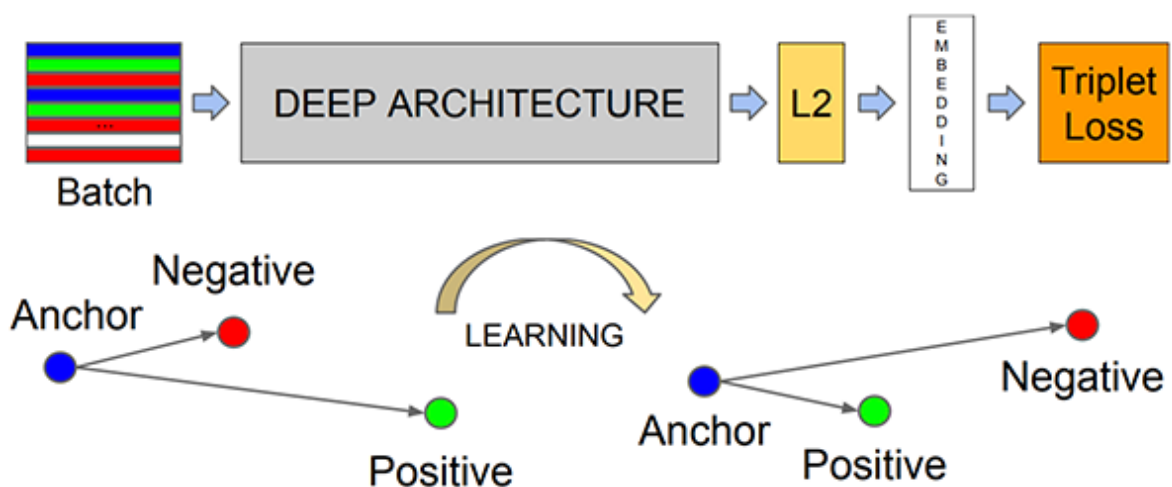


1. To apply *face detection*, which detects the *presence* and location of a face in an image, but does not identify it
2. To extract the 128-d feature vectors (called “embeddings”) that *quantify* each face in an image

First, we input an image or video frame to our face recognition pipeline. Given the input image, we apply face detection to detect the location of a face in the image.

Optionally we can compute facial landmarks, enabling us to preprocess and align the face.

After we've (optionally) applied face alignment and cropping, we pass the input face through our deep neural network:



References :

- [Real Time Face Recognition with OpenCV](#)