**AIML Advanced Computer Vision - Project** 2

# The Real Problem

**What is Face recognition?**

“Say hello to the future” the tagline of iPhoneX marked the advent of face recognition into mainstream apps using it as a feature to unlock the phone. This marks a milestone in itself as far as facial recognition technology is concerned. - Deepa Naik

Facial recognition is a category of [biometric](https://searchsecurity.techtarget.com/definition/biometrics)​[c](https://searchsecurity.techtarget.com/definition/biometrics) software that maps an individual's facial features​ mathematically and stores the data as a face print. The software uses deep learning algorithms to compare a [live capture](https://searchsecurity.techtarget.com/definition/live-capture) ​ or digital image to the stored face print in order to verify an individual's identity.

**Facial recognition is being used in many businesses**

You’re used to unlocking your door with a key, but maybe not with your face. As strange as it sounds, our physical appearances can now verify payments, grant access and improve existing security systems. Protecting physical and digital possessions is a universal concern which benefits everyone unless you’re a cybercriminal or a kleptomaniac of course. Facial biometrics are gradually being applied to more industries, disrupting design, manufacturing, construction, law enforcement, and healthcare.

**Understanding Face Recognition Software**

Face recognition deals with Computer Vision a discipline of Artificial Intelligence and uses techniques of image processing and deep learning. Face recognition algorithms can be further classified based on whether they are used on 2D or 3D images or on finding faces in motion, like in a video.

**Face Detection vs. Face Recognition**

Though they sound similar, the complexity involved in both is vastly different. In Face Detection, the computer recognizes the face within an image and locates its position. If you have used face changer app on Snapchat, you are using face detection. Face recognition deals with identification to establish whose face it is by matching it to an existing face database. You can refer to the images below for the distinction.

## Face detection Face recognition



Apart from identification, other typical features are

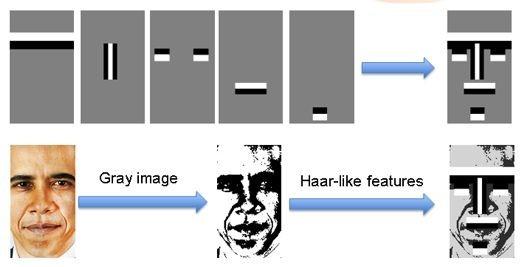
* **Emotion Detection**
* **Age Detection**
* **Gender Detection**
* **Attention Measurement**
* **Sentiment Detection** ● **Ethnicity Detection**



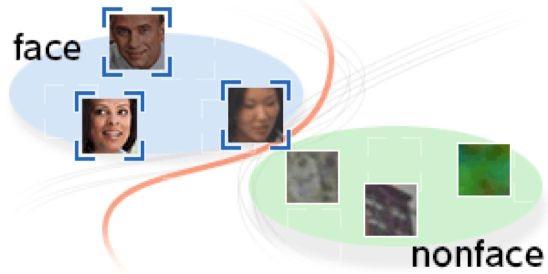
**How Does Face Recognition Work?**

Until the year 2000, there were many different techniques to detect the face, but all were either slow or unreliable, or both. A major change happened in 2001, when Viola and Jones invented Haar-based cascade classifier, a technique used to identify object and it was improved by Lienhart and Maydt in 2002. The result of identifying objects was fast enough (identifying in real-time on normal PC) and was reliable (more than 95% accuracy).

There are two approaches to the facial recognition: Feature-Based Face Recognition and Appearance-Based Face Recognition. Haar-like features are the rectangle which is divided into different rectangles. First, the image is grayscaled, then the haar-like features (rectangle) are shifted through the image, comparing similar image rectangles with Haar-like features, similar ones are marked.



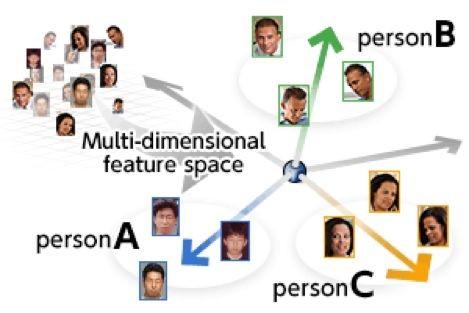
The following are three main steps in the face recognition using haar-like method: Step 1. Detects the “position of the face”



Step 2. Finds the “feature of the face”



Step 3. Search and identify of the “detected face” in the database.



# Project Description

In this hands-on project, the goal is to build a face recognition system, which includes building a face detector to locate the position of a face in an image and a face identification model to recognize whose face it is by matching it to the existing database of faces.

**Face Identification**

In this project, we will build a face identification model by using an existing trained model. The CNN model is taken from the Keras-OpenFace project. The architecture details aren't too important here, it's only useful to know that there is a fully connected layer with 128 hidden units followed by an L2 normalization layer on top of the convolutional base. These two top layers are referred to like the embedding layer from which the 128-dimensional embedding vectors can be obtained. Model training aims to learn an embedding f(x) of image x such that the squared L2 distance between all faces of the same identity is small and the distance between a pair of faces from different identities is large. This can be achieved with a triplet loss L that is minimized when the distance between an anchor image and a positive image (same identity) in embedding space is smaller than the distance between that anchor image and a negative image (different identity) by at least a margin.

To demonstrate face recognition on a custom dataset, a small dataset is used. It consists of around 15-25 face images of 10 different persons. The metadata for each image (file and identity name) are loaded into memory for later processing.

Test the Face Recognition system (the two models together) on a different​ dataset.

**Overview**

In this problem, we use a pre-trained model trained on Face recognition to recognize similar faces.

Here, we are particularly interested in recognizing whether two given faces are of the same person or not. Below are the steps involved in the project.

1. Face Alignment using Dlib

1. Load the dataset and meta information

1. Load the pre-trained model and weights.

1. Generate Embedding vectors for each face in the dataset.

1. Build distance metrics for identifying the distance between two given images.

1. Build SVM classifier to map each image to its right person.

1. Predict using the SVM model.

Instructions for all the above steps are given in the notebook.

**Learn more about Open Face**

<https://blog.algorithmia.com/understanding-facial-recognition-openface/>

Open Face paper:

<http://www.cv-foundation.org/openaccess/content_cvpr_2015/app/1A_089.pdf>

# Project Objectives

The objective of the project is to learn how to do transfer learning and how to use a pre-trained model already trained on the desired task. In this project you are going to use transfer learning to build face detection model and face recognition using pre-trained model The goals of this assignment are as follows:

* Understand how to use pre-trained models
* Using a pre-trained model to do Face Alignment
* Using a pre-trained model to do Face Recognition.
* Learn to use Triplet Loss.
* Learn to use SVM classifier for prediction.
* Learn to combine both the models to test.

**Points: 30 points**

# Reference

Acknowledgment for the datasets.

<http://mmlab.ie.cuhk.edu.hk/projects/WIDERFace/>

# Project submissions

While we encourage peer collaboration and contribution, plagiarism, copying the code from other sources or peers will defeat the purpose of coming to this program. We expect the highest order of ethical behavior.

# Project Evaluation

1. Using load\_weights function to load the given pre-trained weight file
2. Write a function to get the resized cropped out face of an input image from the path given using the face\_bb() function defined above. - 5 Marks
3. Write code to load 2nd and 3rd images in the metadata using load\_image() and show each image and its cropped version side by side for comparison using matplotlib imshow - 7.5 Marks
4. Write code to iterate through metadata and create embedding for each image using nn4\_small2\_pretrained.predict() and store in a list with name `embedding` - 5 marks
5. Plot images and get distance between the pairs given below. --------- 5 Marks
6. Build a 1 layered fully connected neural network using keras and report the accuracy – 7.5 marks
7. Test the classifier

# Project Support

You can clarify your queries related to the project by dropping a mail on Olympus

**Happy Learning!**