# Assignment No:2

Practical Exercise: Facial Recognition using OpenCV and Deep Learning for Binary Classification

## 1) Problem Statement

Implement a facial recognition system using OpenCV and deep learning for binary classification. The system should be able to detect faces in an image and classify them into one of two categories (e.g., known vs unknown, authorized vs unauthorized).

## 2) Libraries Used

Python:  
1. **OpenCV**: For face detection using Haar Cascades and image processing.  
2. **TensorFlow/Keras**: For building and training the binary classification model.  
3. **NumPy**: For array manipulation.  
4. **Matplotlib**: For visualizing the results.

## 3) Theory

Facial recognition involves identifying and verifying individuals based on their facial features. OpenCV's Haar Cascade Classifier can be used for detecting faces in an image, which is a crucial first step before applying deep learning for classification.  
  
Once the faces are detected, a deep learning model can be trained for binary classification (e.g., determining whether a face belongs to a known person or not). Convolutional Neural Networks (CNNs) are commonly used for this purpose, as they can learn spatial hierarchies from images.

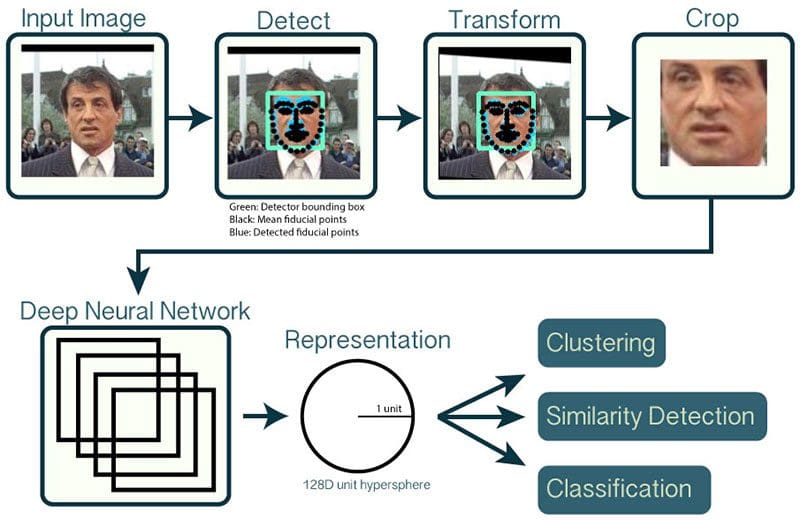
## 4) Methods

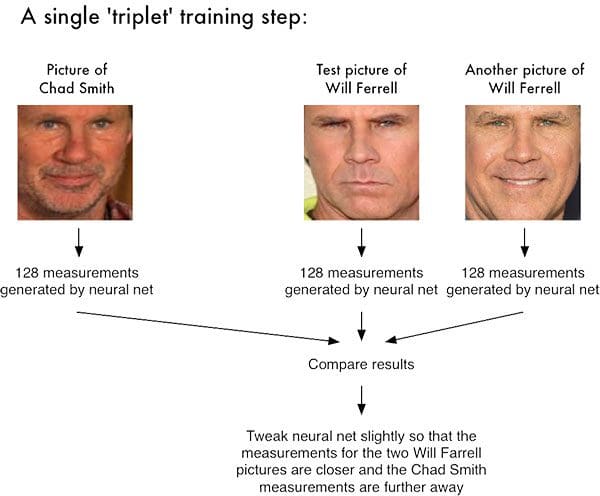
1. **Face Detection**:  
 - The Haar Cascade Classifier in OpenCV is used to detect faces in an image by identifying specific patterns in the pixel intensities.  
 - The `detectMultiScale` function is used to locate faces, and bounding boxes are drawn around each detected face.  
   
2. **Data Preprocessing**:  
 - The detected face images are resized and normalized before feeding them into the deep learning model.  
   
3. **Binary Classification Model**:  
 - A Convolutional Neural Network (CNN) is built using Keras/TensorFlow to classify faces into two categories (e.g., known vs unknown).  
 - The model consists of multiple convolutional and pooling layers, followed by fully connected layers and a softmax activation function for binary classification.  
   
4. **Training and Evaluation**:  
 - The model is trained on labeled face images, with the labels corresponding to the two classes.  
 - After training, the model is evaluated on a test set to measure its accuracy and ability to generalize.

## 5) Advantages and Disadvantages

- **Advantages**:  
 - OpenCV's Haar Cascades are fast and efficient for face detection, making them suitable for real-time applications.  
 - CNNs are effective for image classification tasks, providing high accuracy for facial recognition.  
   
- **Disadvantages**:  
 - Haar Cascades may not perform well under varying lighting conditions or different angles.  
 - The binary classification model requires a sufficient amount of labeled training data to perform well.

## 6) Diagram





## 7) Conclusion

Facial recognition using OpenCV and deep learning provides a powerful solution for real-time face detection and classification tasks. While Haar Cascades are useful for fast face detection, CNNs offer robust performance for binary classification. This combination of tools is widely used in security systems, user authentication, and access control.