Assignment 2

**1. Aim**

To develop a facial recognition system using OpenCV and deep learning for binary classification, capable of distinguishing between two categories: a specific known individual and unknown individuals.

**2. Objectives**

* To understand facial recognition and its implementation using OpenCV.
* To utilize deep learning techniques to classify faces into two categories (binary classification).
* To train and evaluate a model that can identify a person and label other individuals as unknown.
* To create a real-time facial recognition system using a webcam for continuous recognition.

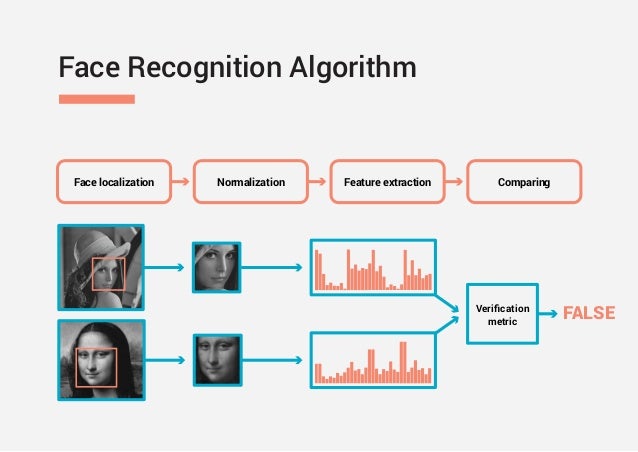
**3. Theory**

Facial recognition is a biometric method of identifying or verifying a person by analyzing their facial features. The process involves detecting a face in an image and classifying it as belonging to a known or unknown category using machine learning algorithms.

* **Binary Classification:** In this context, binary classification refers to the system’s ability to classify faces into two categories: "Known" (specific person) and "Unknown" (other individuals).
* **OpenCV:** OpenCV (Open Source Computer Vision Library) is a popular library for image and video processing. It provides tools like the Haar Cascade Classifier for detecting faces in images.
* **Deep Learning:** In this system, we employ a deep learning-based facial recognition algorithm (such as the Local Binary Pattern Histogram - LBPH) to identify the known individual. LBPH extracts local binary patterns from the face to create a unique histogram for face representation.

**4. Working/Algorithm Used**

The steps involved in implementing facial recognition for binary classification are as follows:



**4.1 Data Preparation:**

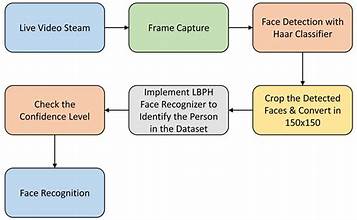
* **Face Detection:**
  + Use the Haar Cascade Classifier from OpenCV to detect faces in grayscale images.
  + The classifier is trained to detect faces using rectangular features from images.
* **Face Data Collection:**
  + A dataset of facial images is collected, where each image belongs to one of two categories: "Known" (e.g., the specific person) or "Unknown" (other individuals).
  + The dataset is organized into folders where each folder represents a label (e.g., "Person1" for the known individual and "Others" for unknown faces).

**4.2 Model Training:**

* **Face Recognition Model (LBPH):**
  + Use the LBPH (Local Binary Pattern Histogram) algorithm, which creates a local binary pattern for each pixel in the image, compares it to its neighbors, and computes histograms to represent faces.
  + The LBPH recognizer in OpenCV is trained on the prepared dataset, where each face is labeled with its corresponding category.

**4.3 Real-Time Recognition:**

* **Real-Time Detection:**
  + A webcam continuously captures video frames.
  + Each frame is processed to detect faces using the Haar Cascade Classifier.
* **Recognition Process:**
  + For each detected face, the LBPH recognizer predicts whether it belongs to the known individual or an unknown person.
  + If the confidence score of the prediction is below a set threshold (e.g., 80%), the person is labeled as "Known," otherwise as "Unknown."
* **Feedback Loop:**
  + The system displays the real-time results by drawing rectangles around detected faces and annotating them with the prediction (e.g., the name of the known person or "Unknown").



**4.4 Algorithm:**

1. Import Libraries (cv2, os, numpy)
2. Load Haar Cascade for Face Detection
3. Define Function to Load Images from Dataset
4. Create LBPH Face Recognizer
5. Start Video Capture from Webcam
6. While Video is Capturing:
   1. Capture Frame from Video
   2. Convert Frame to Grayscale
   3. Detect Faces in Frame
7. For Each Detected Face:
   1. Recognize Face using LBPH Recognizer
   2. Compare Confidence & Identify (Known/Unknown)
   3. Draw Rectangles & Display Name on Frame
8. Display Processed Video Feed
9. Exit on Pressing 'q'
10. Release Video Capture Resources

**Conclusion**

In this project, a facial recognition system was implemented using OpenCV and deep learning-based classification for binary classification. The system efficiently detected faces in real-time using a webcam, and the LBPH model correctly classified them as either "Known" or "Unknown" based on the confidence score. Although the LBPH algorithm performed well, the system can be further improved by using advanced deep learning techniques, such as Convolutional Neural Networks (CNNs), for better accuracy and reliability.