

# Advanced Face Recognition System with Real-Time Detection

## Project Domain / Category

Image Processing.

### Abstract

This project aims to develop a comprehensive face recognition system using machine learning that not only detects and recognizes faces but also saves captured face images in a database for future reference. Additionally, the system will feature a user interface to manage the database, allowing users to upload new images, view stored data, and manage the system. The core of the system will rely on deep learning techniques for face recognition and database integration to store and retrieve face data efficiently.'

### Functional Requirements:

**1. Collect and Prepare Data:** Collect face images either from public datasets (e.g., LFW, CelebA) or by capturing faces via webcam. Preprocess the images by resizing, aligning, and normalizing them.

**2. Develop the Face Recognition System:** Implement face detection using OpenCV or Dlib. Use pre-trained models like FaceNet to extract face embeddings from detected faces. Train or fine-tune machine learning models (e.g., KNN, SVM) using the extracted embeddings for recognition tasks.

**3. Database Integration:** Create a database schema in SQLite or MySQL to store: Face images, Face embeddings, Metadata (e.g., name, ID). Implement functions to store, retrieve, update, and delete records from the database.

**4. Build the User Interface:** For Desktop (Tkinter): Create an interface where users can view all stored face images along with names and IDs. Implement buttons to upload new face images and save them to the database. Add features to edit or delete face entries. For Web (Flask/Django): Develop a web-based interface to manage the face database, allowing remote access.

**5. Real-Time Face Recognition:** Capture live image streams using a webcam. Detect and recognize faces from the image feed by comparing the extracted embeddings with the stored database. Display the recognized name or ID on the image. Allow users to search for specific individuals in the database by name or ID and display their corresponding face images and embeddings. Enable users to modify or delete stored face entries (e.g., change name, remove a face).

The enhanced face recognition system will not only detect and recognize faces but will also allow for efficient storage and management of face data in a database. The added interface will provide users with an easy-to-use platform to manage face images and metadata, making it suitable for real-world applications like security, attendance tracking, and access control systems.

### Tools and Technologies Required

**1. Programming Language:** Python: The primary language for the implementation of machine learning algorithms and the database interface.

**2. Libraries:** OpenCV for real-time video processing, face detection, and webcam integration. Dlib for advanced face detection and feature extraction. TensorFlow / Keras for building deep

learning models (CNN, FaceNet, VGGFace). Flask/Django for web-based UI (optional). Tkinter for desktop-based GUI. SQLite / MySQL for database management.

3. **Face Recognition Libraries:** FaceNet, VGGFace, or Dlib for extracting face embeddings. Scikit-learn for classifiers (e.g., KNN, SVM).

4. **Database:** SQLite or MySQL for managing face images, metadata, and embeddings.

5. **Webcam or External Camera:** For capturing real-time video streams and detecting faces.

6. **IDE:** PyCharm, VS Code, or Jupyter Notebook for coding and testing the system.

### Face Recognition Dataset Links

1. [Labeled Faces in the Wild (LFW)](<http://vis-www.cs.umass.edu/lfw/>)
2. [CelebA Dataset](<http://mmlab.ie.cuhk.edu.hk/projects/CelebA.html>)
3. [YouTube Faces Database](<https://www.cs.tau.ac.il/~wolf/ytfaces/>)
4. [VGGFace2 Dataset]([http://www.robots.ox.ac.uk/~vgg/data/vgg\\_face2/](http://www.robots.ox.ac.uk/~vgg/data/vgg_face2/))
5. [WIDER Face Dataset](<http://shuoyang1213.me/WIDERFACE/>)

Supervisor:

Name: Madiha Faqir Hussain

Email ID: [madiha.hussain@vu.edu.pk](mailto:madiha.hussain@vu.edu.pk)

Skype ID: madiha.akhtar74