SOC'25 - The Eyes of the Machine: A Journey into Face Detection

Jayent Dev
Web and Coding Club, IIT Bombay

June 2025

Abstract

This report documents the ongoing work done under the SOC'25 (Summer of Code 2025) program by the Web and Coding Club, IIT Bombay. The project is centered around building a facial recognition system using a Siamese Neural Network architecture, Python, and open-source libraries like TensorFlow and OpenCV. This report covers both the foundational concepts behind facial recognition and the learning journey undertaken to build the system. Future applications such as attendance systems and verification services are also considered.

Contents

1 Introduction

Facial recognition is the process of identifying or verifying an individual based on their facial features from images or video frames. With increasing advances in deep learning and computer vision, face recognition has moved from being a complex academic problem to a real-world utility across various domains.

This project was undertaken as part of the SOC'25 program, with a learning-first approach and focus on foundational understanding and clean implementation.

2 Timeline and Learning Process

Duration: 24th May 2025 – 14th July 2025

2.1 CS101 Revision

The project began with revision of algorithmic basics learned in CS101. Revisiting concepts such as loops, conditionals, and array manipulations helped solidify programming logic, essential for image processing and machine learning workflows.

Slides from CS101 were revisited and stored in the GitHub repository.

2.2 Learning Python

Since Python is the primary language for AI and ML workflows, extensive time was dedicated to learning it. Two playlists were followed:

- Python Programming [English]
- Python Programming [Hindi]

2.3 Learning Core Libraries

To manipulate arrays, handle datasets, and visualize images and features, the following libraries were studied:

- **NumPy**: For numerical computations.
- **Pandas**: For tabular data handling.
- **Matplotlib**: For plotting and debugging.

A custom document on Python modules was created and shared in the repository: Python Libraries Documentation

3 Overview of Facial Recognition

Facial recognition involves several sequential tasks:

1. **Face Detection**: Locating faces in an image.
2. **Face Alignment**: Normalizing scale, tilt, and rotation.
3. **Feature Extraction**: Converting facial image into numerical vectors.
4. **Recognition**: Comparing feature vectors for matching or verification.

Each step is critical for the accuracy and reliability of the system.

4 Proposed Architecture: Siamese Neural Networks

A Siamese Neural Network is a type of neural network architecture that learns similarity between inputs rather than direct classification. It works by learning a shared feature space and computing a distance between two inputs.

- Two identical subnetworks (typically CNNs) process two input images.
- The outputs are embeddings compared using a distance metric.
- Training uses contrastive or triplet loss to minimize distance for same-person pairs and maximize for different-person pairs.

Refer to the official document: Siamese Neural Network PDF

This approach is ideal for "one-shot learning" tasks where a new identity needs to be recognized using very few examples.

5 Libraries and Tools

5.1 TensorFlow

TensorFlow provides a flexible environment to build and train deep learning models like Siamese Networks.

5.2 OpenCV

Used for:

- Image preprocessing (resize, grayscale, histogram equalization)
- Real-time webcam feed and face detection
- Bounding boxes and annotation

5.3 face_recognition

For quicker prototyping, Dlib-based models like the `face_recognition` library may be used initially before custom model training.

6 Current Repository Status

At this stage, the GitHub repository includes:

- Learning resources: Python doc, Siamese network PDF, CS101 slides.

Code development will start soon as till now it was just building up the basics for further making of the program.

7 Challenges and Road Ahead

Challenges Expected:

- Low-light and occlusion handling.
- Bias in training data.
- Performance optimization for real-time use.

Planned Milestones:

1. Complete dataset preparation and augmentation.
2. Build and train Siamese Network model.
3. Integrate live camera feed for real-time verification.
4. Export embeddings and match faces with database.

8 Ethical Considerations

Although this is a personal learning project, any future deployment of facial recognition requires attention to:

- Data privacy and consent
- Transparency in decision-making
- Accuracy across demographics
- Proper logging and accountability

9 Future Applications

Once complete, the same system can be adapted for:

- **Automated Attendance** – Face-based logging in classes or offices.
- **Security Verification** – Unlocking systems via face match.
- **Visitor Management** – Recognizing repeat visitors in smart buildings.

These use cases are not currently being implemented but remain potential applications for the underlying architecture.

10 Conclusion

This project aims to demystify facial recognition by learning, implementing, and testing a system from scratch. With a step-by-step learning approach and proper modular design, the goal is to produce a working prototype by July 2025. The report presented here is a mid-way technical summary of both foundational theory and learning journey.

References

- Schroff, F., Kalenichenko, D., Philbin, J. (2015). FaceNet.
- Koch, G., Zemel, R., Salakhutdinov, R. (2015). Siamese Neural Networks for One-shot Image Recognition.
- `https://opencv.org`, `https://www.tensorflow.org`