Real-Time Face Re-Identification using YOLO and FaceNet

Shashank Reddy and Prisha Khairnar

Abstract

This project presents a robust real-time system for face detection, identification, and re-identification in live video streams. The architecture integrates **YOLOv3** for high-speed and accurate face detection, and **FaceNet** for generating compact and discriminative facial embeddings. A tracking mechanism using OpenCV's **CSRT tracker** ensures continuous face tracking across frames. When tracking confidence drops, **re-detection and re-identification** are triggered using cosine similarity and contextual heuristics. The system has been tested under various lighting and occlusion scenarios, demonstrating reliable performance.

1. Introduction

Face recognition technologies are increasingly relevant in areas like surveillance, attendance systems, and human-computer interaction. A particular challenge arises in real-time environments: maintaining the identity of a person even after temporary occlusion or disappearance from the camera frame. This project was designed to address that challenge by building an end-to-end pipeline for real-time **face re-identification**.

We started by experimenting with traditional techniques and gradually transitioned to deep learning-based methods to improve robustness and accuracy.

2. Initial Approaches

2.1 Traditional Methods

We began with conventional approaches that required low computational resources but lacked robustness:

- **Haar Cascades**: Basic detection model; failed for side profiles and non-frontal faces.

- **CLAHE and Histogram Equalization**: Applied for low-light enhancement, but results were inconsistent.
- **Dlib HOG + SVM Detector**: Showed improvement over Haar cascades, but performed poorly with complex lighting and occlusion.

2.2 Mediapipe

We experimented with **Google's Mediapipe face detection** framework, which showed promise due to its lightweight architecture. However, it suffered from significant inconsistencies under varied lighting and head pose conditions, prompting the shift towards deep learning-based detection.

3. Final Pipeline

3.1 Detection with YOLOv3

We use **YOLOv3**, pretrained on the **WIDER Face dataset**, for real-time detection of human faces. YOLOv3 was selected for its:

- High detection speed (real-time capable)
- Robust performance across a range of facial orientations and lighting conditions

Post-processing includes **Non-Maximum Suppression (NMS)** to eliminate overlapping detections.

3.2 Embedding with FaceNet

Initially, we used **Dlib's facial embedding model**, but it provided subpar accuracy in identity matching. This led us to adopt **FaceNet**, which generates **128-dimensional embeddings** that are both compact and discriminative.

To create a reference for each detected person, embeddings are **averaged over a few frames** and stored for future matching.

3.3 Face Matching

Face re-identification is carried out using **cosine similarity** between current and stored embeddings. To enhance robustness, we introduce a **composite similarity score** that combines:

- Cosine similarity (weight: 0.8)
- Positional similarity based on the last known location of the face (weight: 0.2)

3.4 Light Adjustment

Handling varying lighting is crucial in real-world environments. Our system includes:

- **Gamma Correction**: Applied dynamically with a gamma value of 1.7
- **Conditional Denoising**: Utilizes **Non-Local Means Denoising** for frames where brightness falls below a predefined threshold

3.5 Tracking Mechanism

To maintain the identity of a face across frames, we implemented a tracking system using OpenCV's **CSRT tracker**. CSRT is robust to scale changes and occlusions, making it suitable for real-time tracking.

Prior to finalizing CSRT, we also experimented with the **Multi-Channel Perception (MCP)** tracker, which was less consistent in handling jitter and motion blur.

The tracking pipeline works as follows:

- A face is selected and tracked using CSRT.
- If the confidence score drops or the tracker fails, the system initiates **re-detection and re-identification**.
- The new candidate face is matched against existing identities using the composite score.

4. Re-Detection Heuristics

When a face is lost (e.g., due to occlusion or leaving the frame), re-detection is triggered. To decide whether a newly detected face corresponds to a previously tracked identity, we apply the following heuristics:

- **Brightness Analysis**: The detected face's brightness is used to set dynamic thresholds for similarity.
- **Composite Score Calculation**: The cosine and positional scores are combined to match the new face to known identities.

- **Temporal Smoothing**: Averaging embeddings over time helps reduce noise and misidentification during re-detection.

5. Results

Our system successfully demonstrates real-time performance under varied conditions. Key outcomes include:

- Accurate face detection and tracking across lighting conditions
- Successful re-identification of faces even after brief occlusions
- High reliability in low-light scenarios, due to gamma correction and denoising
- Embedding stability using FaceNet, significantly improving matching over Dlib

6. Known Limitations

- If a different person suddenly appears very close to the camera or in front of the tracked face, the tracker may incorrectly switch to the new face.
- Multiple similar-looking faces in a frame can still cause minor misidentifications due to visual overlap.
- The system currently does not use facial landmarks or head pose estimation, which could further improve re-identification accuracy.

7. Conclusion

This project successfully implements a real-time face re-identification system using a combination of YOLOv3, FaceNet, and CSRT tracking. The pipeline demonstrates reliability, efficiency, and robustness under varied real-world conditions. Future improvements may include:

- Occlusion-aware models
- Incorporating facial landmarks
- Enhanced tracker switching logic using identity confidence history

Credits

Developed by: Shashank Reddy and Prisha Khairnar
Motivation: Final task assigned by Rugved