Introduction

This documentation serves as a comprehensive guide for setting up and running a face recognition system that utilizes a dataset of images, including a special focus on images from the TV series "The Big Bang Theory." Designed to cater to beginners and experts alike, it outlines the environment setup, step-by-step user installation instructions, how to run the code with simple examples, and dives into more technical details of the implementation. By following this guide, users can efficiently encode faces from a dataset and recognize faces in both images and videos using a combination of OpenCV, Dlib, and CNN technologies.

Diagram/Image/Video/Demo

To better understand the structure and flow of the face recognition system, a diagram or video demonstration would be beneficial. However, such a demo is not included in this document and is suggested as a future improvement.

User Installation Instructions

Prepare Your Environment: Before starting, ensure you have Python installed on your machine. This project requires Python 3.x.

Clone the Repository: Download the code repository to your local machine.

Install Dependencies: Navigate to the project directory in your terminal and run pip install -r requirements.txt to install the necessary libraries listed in the requirements.txt file.

How to Run the Code: A Simple Example

Encoding Faces: Run the encode\_faces.py script to encode the faces present in the dataset. This script processes each image, detecting faces and creating a unique 128-dimensional embedding for each detected face.

Command: python encode\_faces.py

Recognizing Faces in Video: To recognize faces in a video, run the recognize\_faces\_video.py script. Before running, customize the script by setting the output filename on line 29 and the input filename on line 36.

Command: python recognize\_faces\_video.py

Optional - Recognizing Faces in Images: For recognizing faces in static images, use the recognize\_faces\_image.py script. Input and output filenames can be set on lines 28 and 86, respectively.

Command: python recognize\_faces\_image.py

More Technical Details

The core of this face recognition system is built upon the integration of OpenCV, Dlib, and CNNs to identify faces within videos and images. Here's a brief overview of the technical process:

Image Preprocessing: Includes brightness enhancement, histogram equalization, and contrast enhancement to prepare images for detection.

Face Detection: Utilizes a CNN model to detect faces in images by learning and extracting important features through convolutional and pooling layers.

Encoding Faces: Each detected face is encoded into a 128-dimensional embedding, which is used to recognize the face among a database of known faces.

Recognition Process: Implements a matching algorithm to find the most similar face embeddings and identifies the person based on the highest match percentage.

Known Issues/Future Improvements

Demo Inclusion: A future version of this documentation should include a diagram or video demonstration to provide clearer context and understanding.

Enhanced Recognition Accuracy: Improvements in the face encoding and matching algorithms could enhance the recognition accuracy, especially in varying lighting conditions and angles.

Scalability: The system's performance with significantly larger datasets needs evaluation and potential optimization to handle more extensive image libraries efficiently.

This guide aims to provide a solid foundation for setting up and utilizing a face recognition system. As technology advances, it's expected that future iterations will address current limitations and introduce new features for an improved user experience.

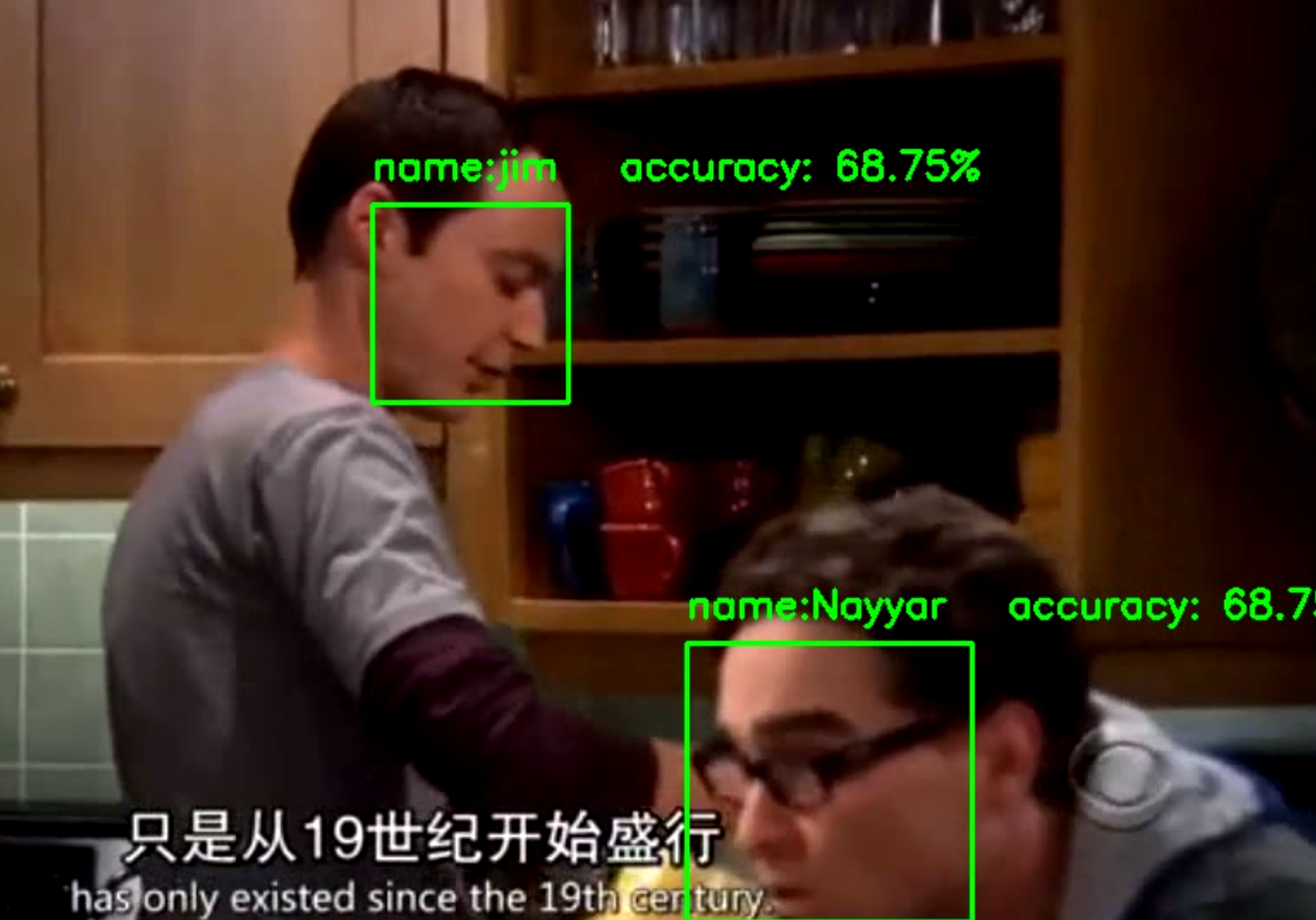


Figure 1. Output of the code