Department of Electrical and Computer Engineering
North South University (NSU)

CSE 440: Artificial Intelligence
Section 05

Project

Total Marks: 100
Proposal (20) + Execution (60) + Presentation (20)

Instructor: Dr. Mohammad Mahmudul Alam Semester: Spring 2025

Title:	End to end Object Detection Project
Name:	Abir Md. Fuad Hasan Anan
NSU ID:	1931120642

Proposal

Title: End to end Object Detection Project

Introduction:
The project is about creating a realtime object detection software which will allow us to detect any object it sees. The project is built using YOLOv5 (You Only Look Once), a real time object detection model powered by PyTorch. A custom dataset is collected using a Mac M1 camera, and the training pipeline is managed within a Conda environment, I will use Roboflow for dataset preprocessing and augmentation and annotation.

Problem Statement:
The problem of visually impaired people and the problem of deaf people who uses sign languages to communicate and system that are used in shops to or roads to determine objects will be more easy and accessible, AI realtime object detection can help us in a lot of sectors by integrating this API into a existing system. I will train more data into the model as days go by and make the accuracy level higher

Objectives:
The key objectives of this **End to end object detection** project are:

- Identify and analyze the problem: Understand the challenges in recognizing and interpreting sign language gestures using computer vision techniques. Analyze existing solutions and their limitations.
- Develop a working solution or prototype: Build a deep learning-based object detection model using **YOLOv5 and PyTorch** to accurately detect and classify sign language gestures.
- Improve efficiency or usability in the target domain: Optimize the model for real-time performance, ensuring low latency and high accuracy for practical applications such as assistive tools and communication aids.
- Validate the solution through testing and analysis: Train the model on a custom dataset, evaluate its performance using metrics such as **mAP (mean Average Precision), recall, and F1-score**, and refine it based on results.
- Document findings and present conclusions: Maintain a detailed report of data collection, preprocessing, training, and evaluation, ensuring reproducibility and providing insights for future improvements.

Methods:

1. I will be using VS code for the main compiler of the project.
2. My chosen language for the project is python.
3. At first I will be using Git hub and create a public repository for the project then I will be updating it as soon the work objectives are completed.
4. I will write the root code in python using VS code. Then I will be using a Data Collector python script to collect my camera input.
5. I will use Roboflow to annotate and augment the data.
6. Then I will be deploying those unique ID data to my main script.
7. I will use YOLOv5 to create the project

Datasets and Tools:

For the dataset I will create my own custom labelled dataset for first few project updates. Then I will use existing models to train my project using the dataset.
To create custom dataset I will be using

1. Roboflow
2. LabelImg
3. Google Colab

For tools I will be using

1. Microsoft VS code
2. M1 Mac air
3. Conda environment
4. YoLo v5 Pytorch
5. Git Hub to create a repository

Project Update

Title: End to end Object Detection Project

Abstract:
For the past few weeks I have been working on the project as one and only contributor. The project solves the problem for smart surveillance, helps to recognize objects for visually impaired, will help to accomplish autonomous driving and sign language converter. I have gathered all the knowledges and tools to finish the work within given period of time. So far I have been working on the Sign Language Part of the project.

Methodology:
I had to start the task from the scratch and by now I have accomplished a certain amount of goals which are given below with screenshots as proof.

1. I have created a Github repository for the whole project to store the data and works I am doing on the project. The link : https://github.com/fuadhasan247/End-to-end-Object-Detection-Project.git

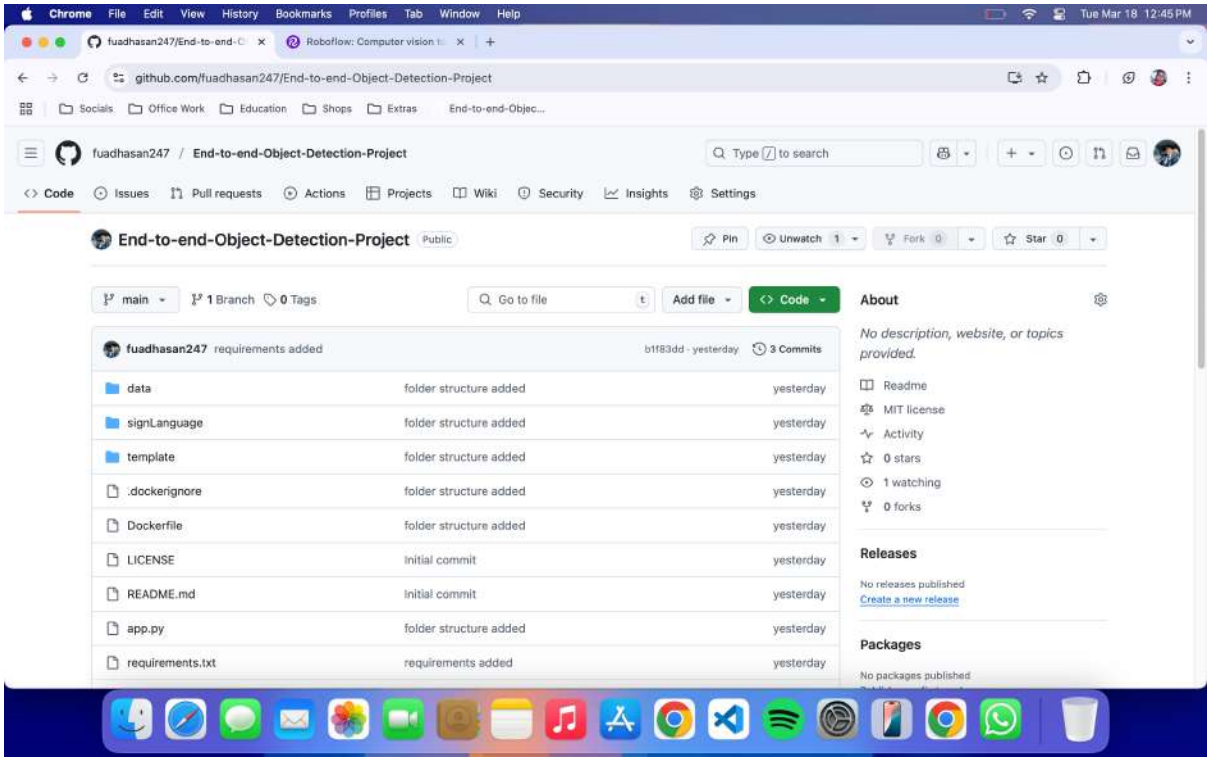


Fig 1: Github repository

2. I have wrote some codes for template, main app, and data collector using python. The template structures the whole project in Github, the main app contains the debug part of the system and to train models I have used the data collector.

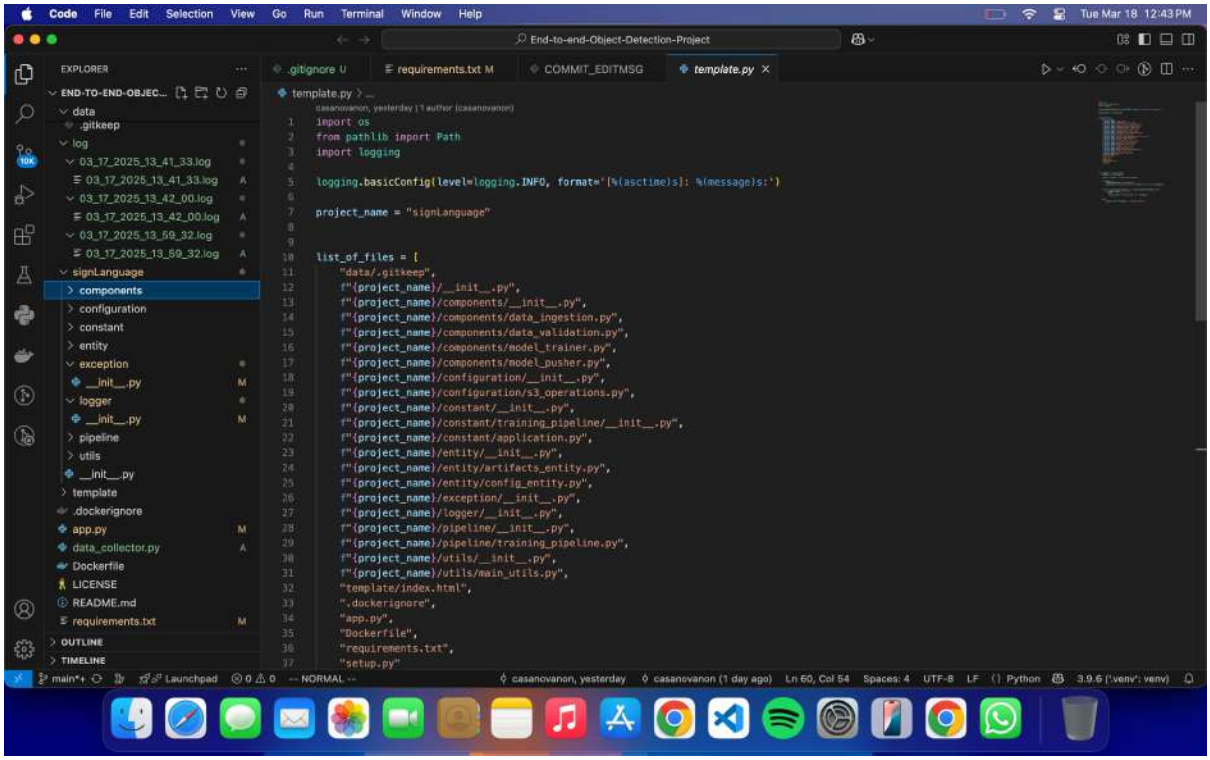


Fig 2: Template.py

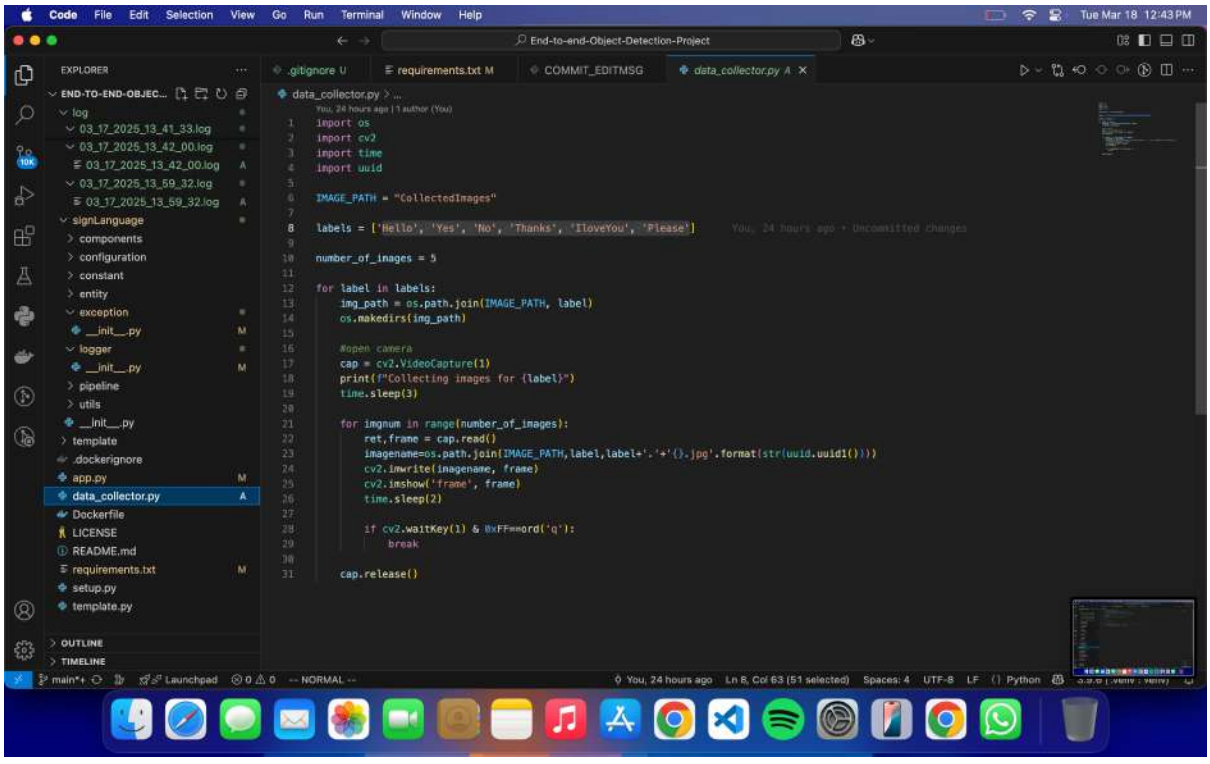


Fig 4: Data_collector.py

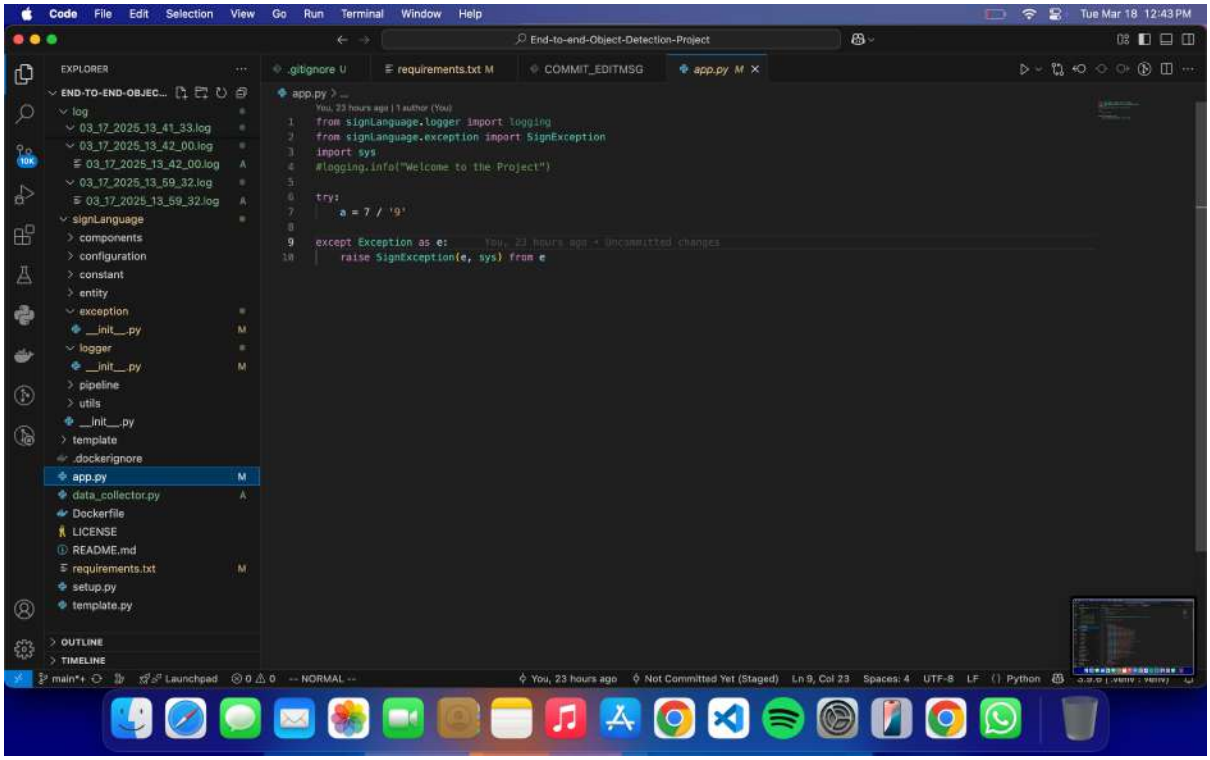


Fig 3: App.py

3. I have collected 30 samples of data using my Laptops camera. In the data collector script, I have created all the images to have an unique Name and ID.

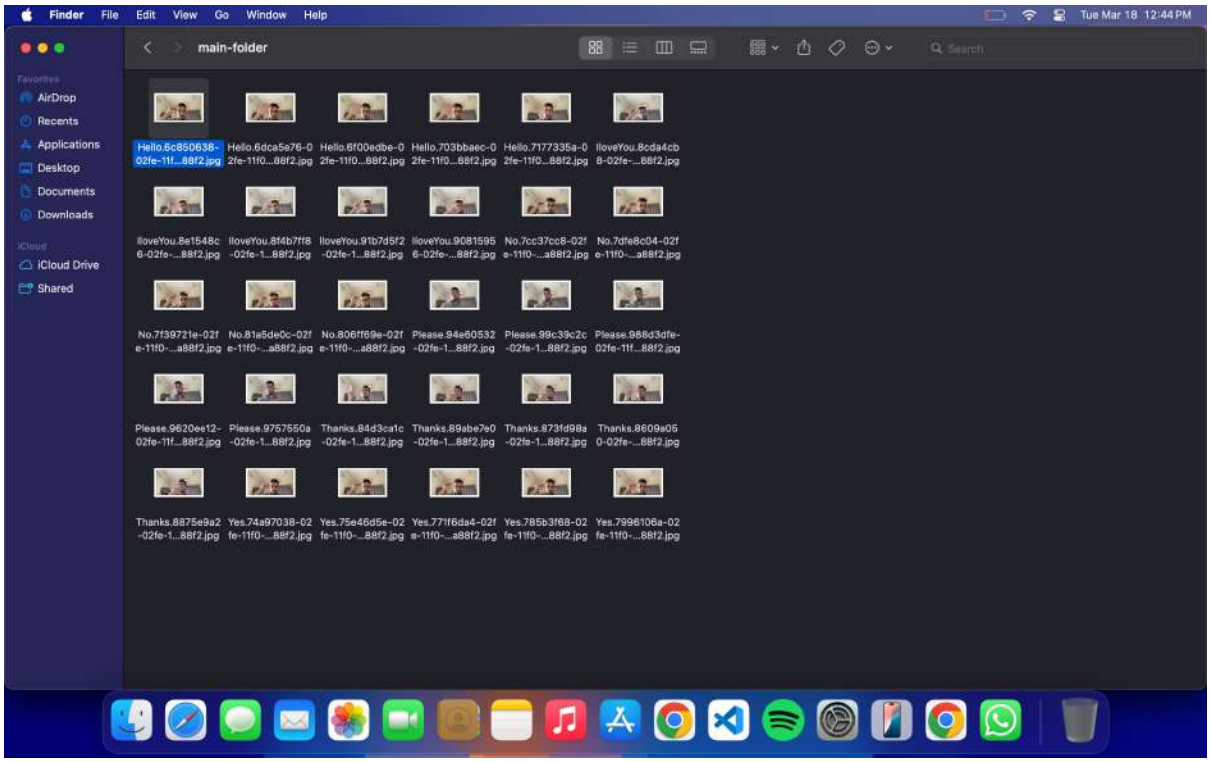


Fig 5: Collected Data

4. I have successfully annotated all the collected pictures in RoboFlow to train the models.

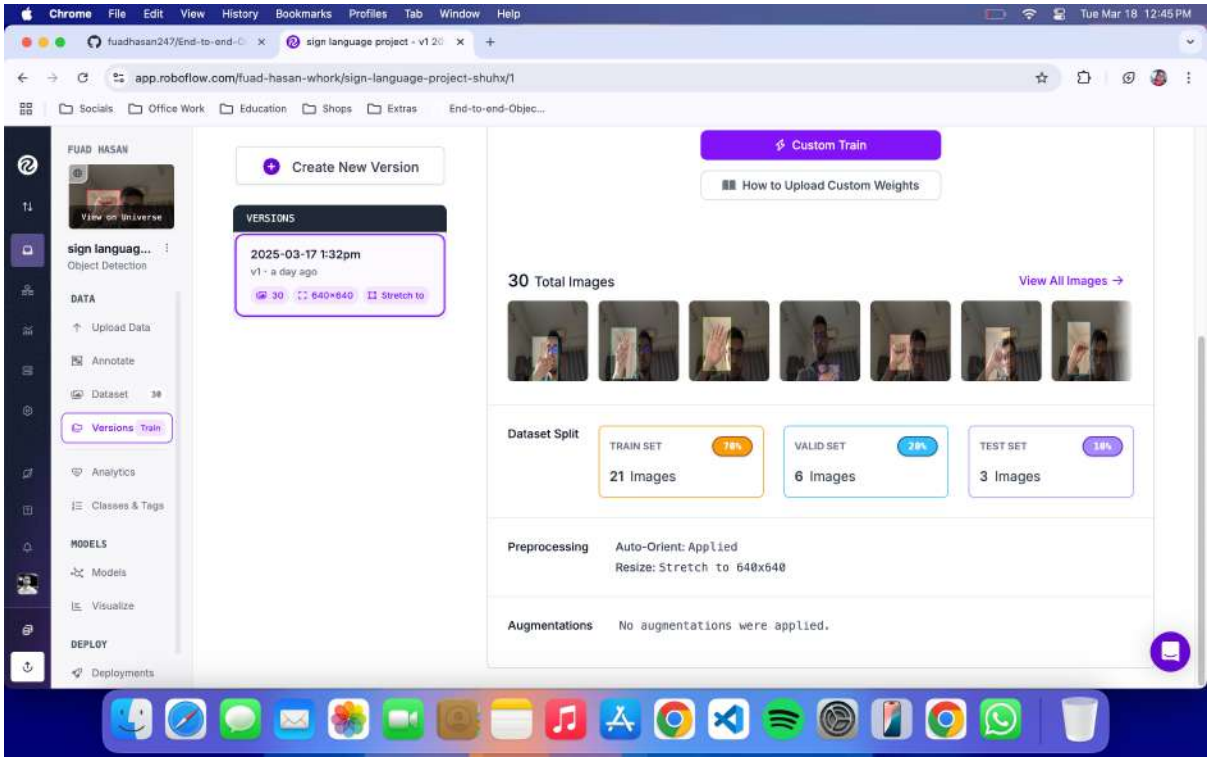


Fig 6: RoboFlow

Initial Results:

As of now, I have everything synced in my Github repository. My Master app can write logging information with time stamps. It can also handle exceptions and my data collector script can also collect data using my camera hardware. The main master folder of the project looks like this till now.

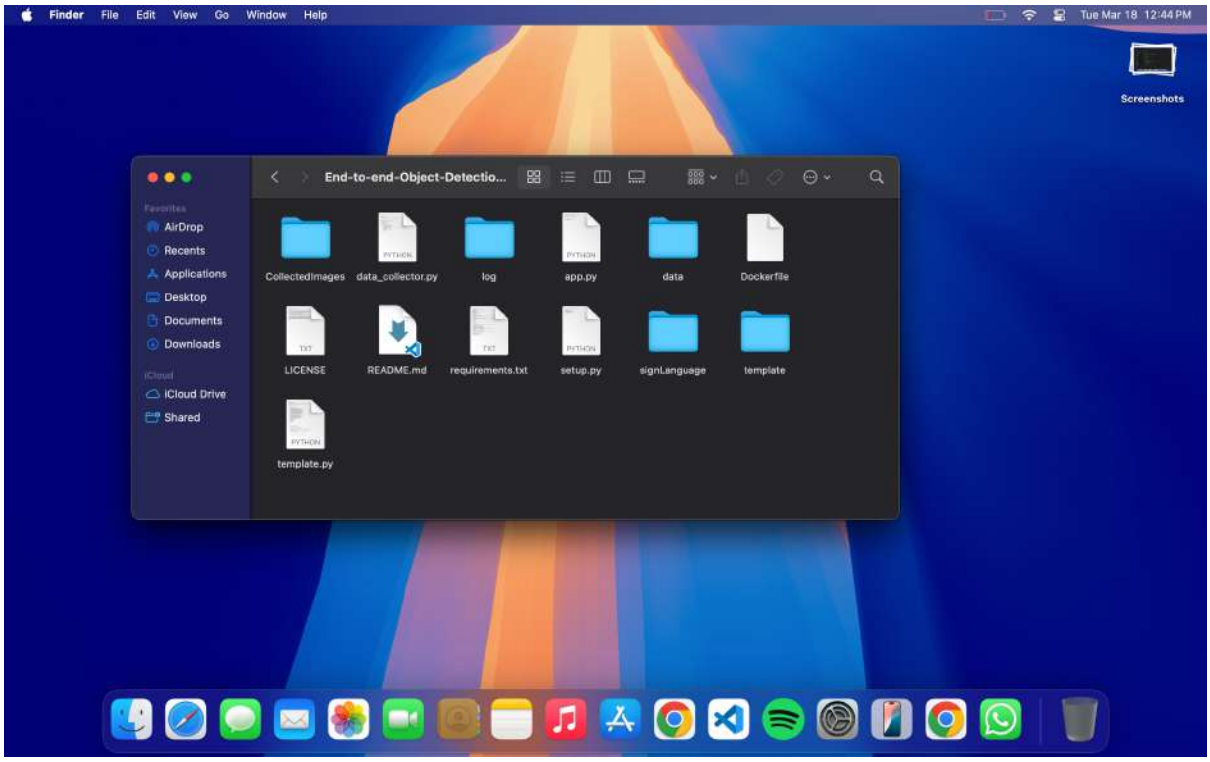


Fig 7: Root folder

Next Steps:

For the next step I will be deploying the master app, finish writing all the code and scripts by doing these steps.

1. I will finish writing the master app code
2. Train more custom data
3. Add an existing data set for the variety of data
4. Deploy the master app in GitHub and finalize the whole project

Final Project

Title: End-to-end Object Detection Project

Abstract:
The main objective of this work involves creating a complete system to detect sign language hand movements through deep learning models. The system makes use of YOLOv5 together with PyTorch to identify hand signs collected through a Mac M1 camera using Roboflow for data preprocessing and collection. This project develops a highly responsive solution for hearing-impaired accessibility which accurately and efficiently interprets sign language to improve communication for the entire community.

Problem analysis and dataset creation take place first in the project flow followed by model training for validation followed by performance evaluation. The project benefits from using Conda-based environments and VS Code for development because this platform approach provides flexible and reproducible operations in the pipeline. The main purpose is to advance assistive technology while creating equipment which can be used in translation software and educational systems alongside accessibility programs. The proposed project demonstrates strong potential to help users with sign language needs achieve better digital inclusivity through enhanced accessibility.

Introduction:
Human beings need communication as a basic requirement and sign language operates as a necessary method for deaf and hard-of-hearing people to express themselves. The lack of communication essential between users of sign language and people without sign language understanding frequently results in social and interpretive difficulties. The development of artificial intelligence (AI) and computer vision technologies along with their fast growth now creates new prospects to resolve communication challenges using automated sign language detection systems.
The project establishes a complete object detection solution for sign language detection that uses current deep learning frameworks to identify hand gestures during real-time operation. This system employs YOLOv5 as its object detection foundation because it runs on PyTorch framework to deliver real-world appropriate speed and accuracy balance. The planned development environment relies on Conda together with Visual Studio Code for efficient modular management of experiment processes. The sign language data collection happens through Mac M1 cameras to obtain high-quality images and Roboflow handles all dataset processes from annotation through preprocessing and augmentation.
This project carries major importance that goes further than basic technical advancements. Real-time sign language interpretation through this system gives

machines the ability to accomplish better inclusivity and enhanced accessibility and empowered communication throughout diverse communities. Sign language detection technology presents meaningful potential for the application domain which includes education and healthcare alongside customer service and public delivery.

This research aims to develop foundational capabilities for intelligent systems to understand non-verbal communication because it supports the main objective of human-centered technology accessible to all communities.

Problem Statement:

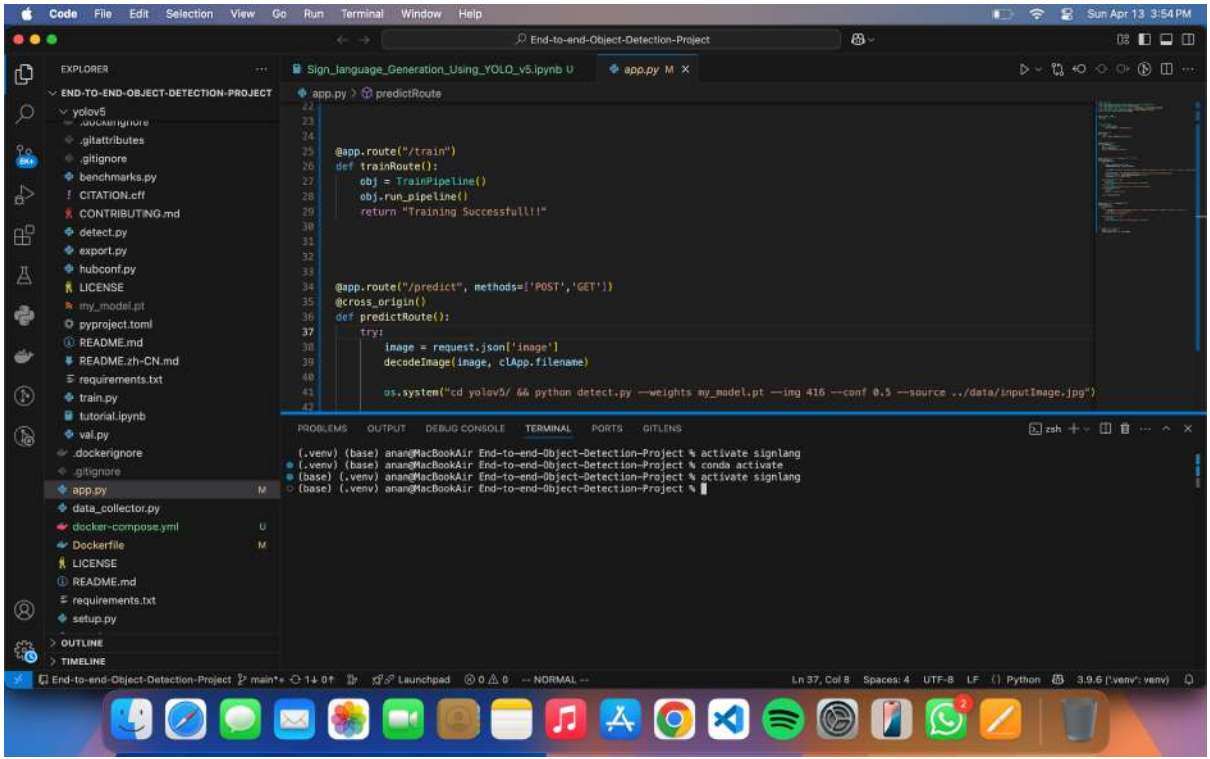
Sign language functions as the main communication method for deaf and hard hearing populations yet technical mainstream systems provide inadequate support to this sector. The absence of high-quality sign language interpretation systems blocks communication paths between people who use sign language and everyone else. The present solutions work with big equipment and operate through restricted gesture systems and cannot succeed in everyday practical applications.

The development of an efficient user-friendly scalable system needs to detect sign language gestures through real-time interpretation using standard camera feeds. The main difficulty involves detecting many different hand gestures correctly when lighting varies because of different backgrounds and different hand morphologies while also maintaining both fast operation and user convenience.

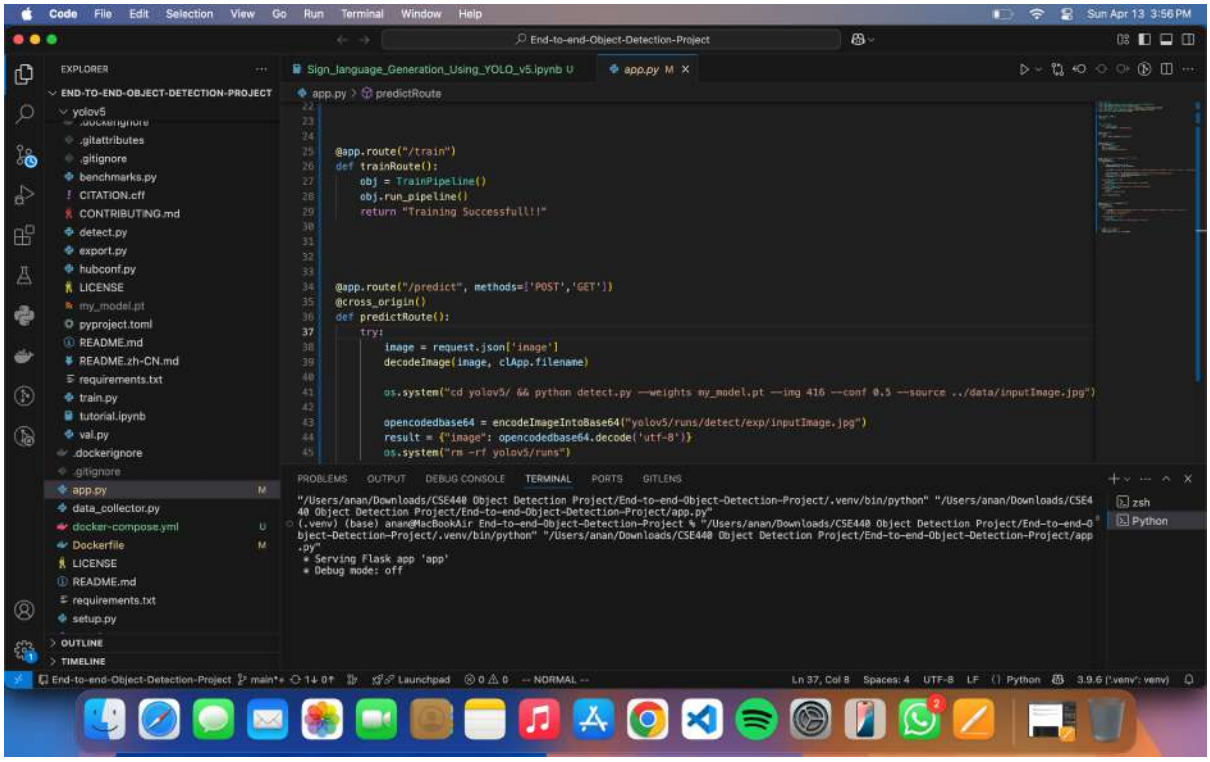
The system tackles this problem through its development of an end-to-end object detection solution based on YOLOv5 and PyTorch that detects sign language gestures in live video entries. This solution functions to close communication gaps so users can achieve more inclusive digital accessibility.

Methodology:

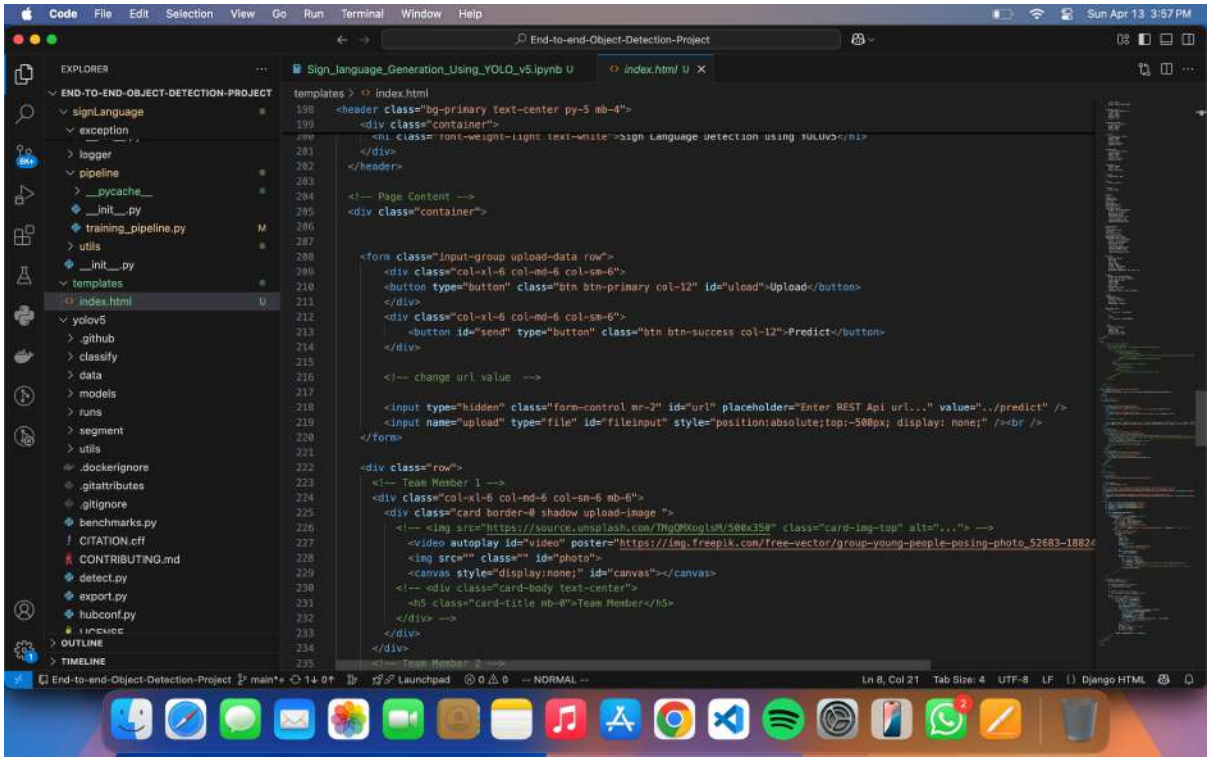
1. At first I have created a virtual environment in my system which is SignLang. Which needs to be activated before hand using linux shell command “activate signlang”



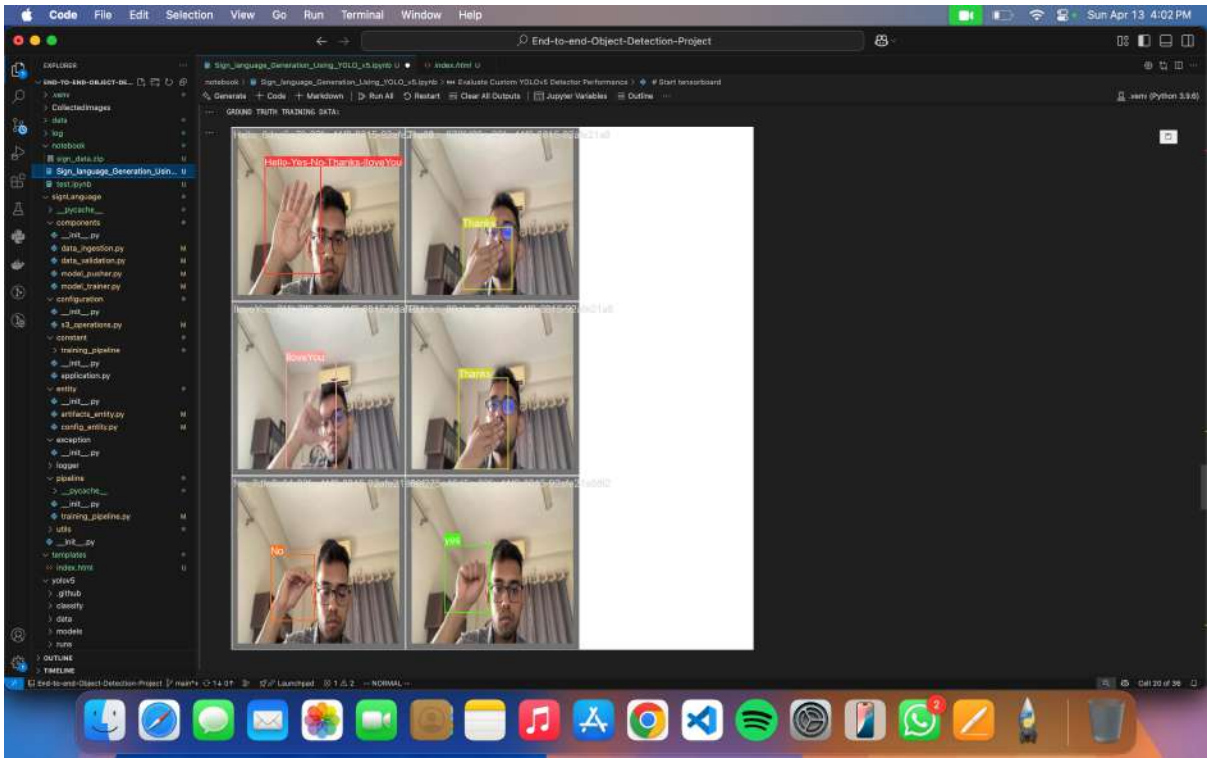
2. Then I have written an App.py to execute the internal local server to run the yoloV5 model.

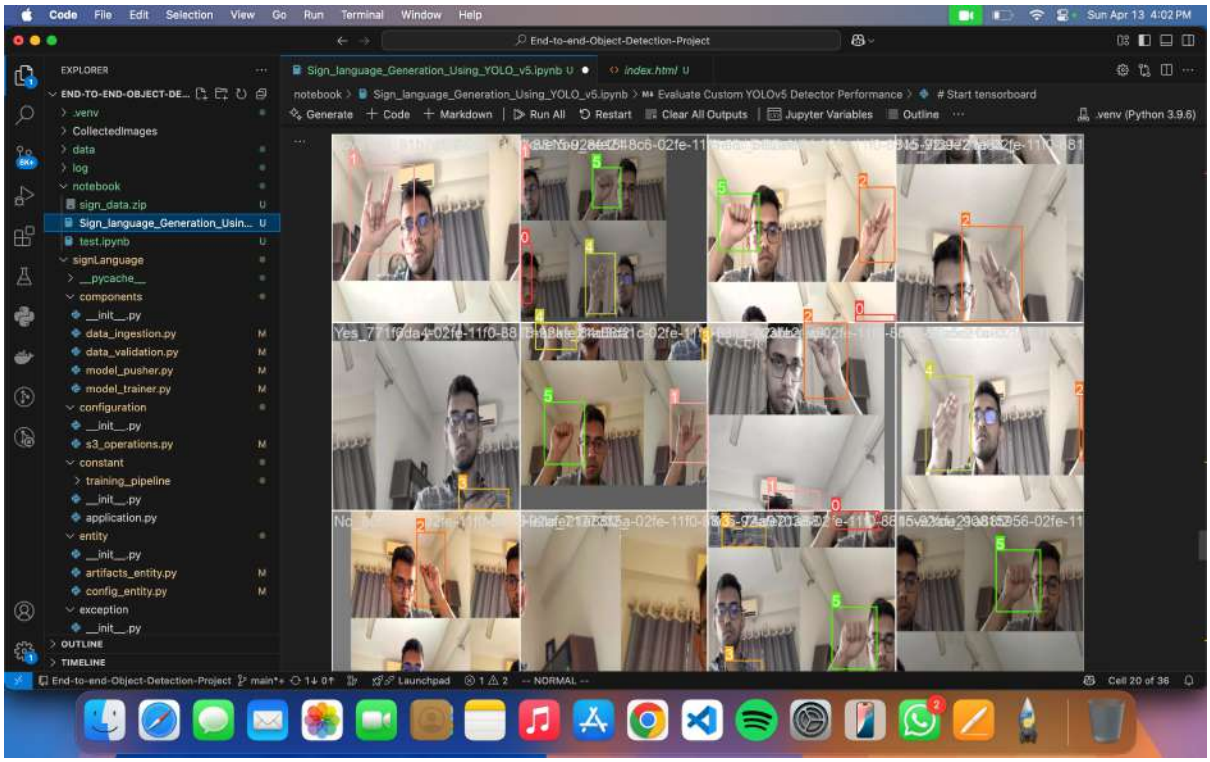


3. I have created a HTML webpage which runs locally to turn on my camera and start detection

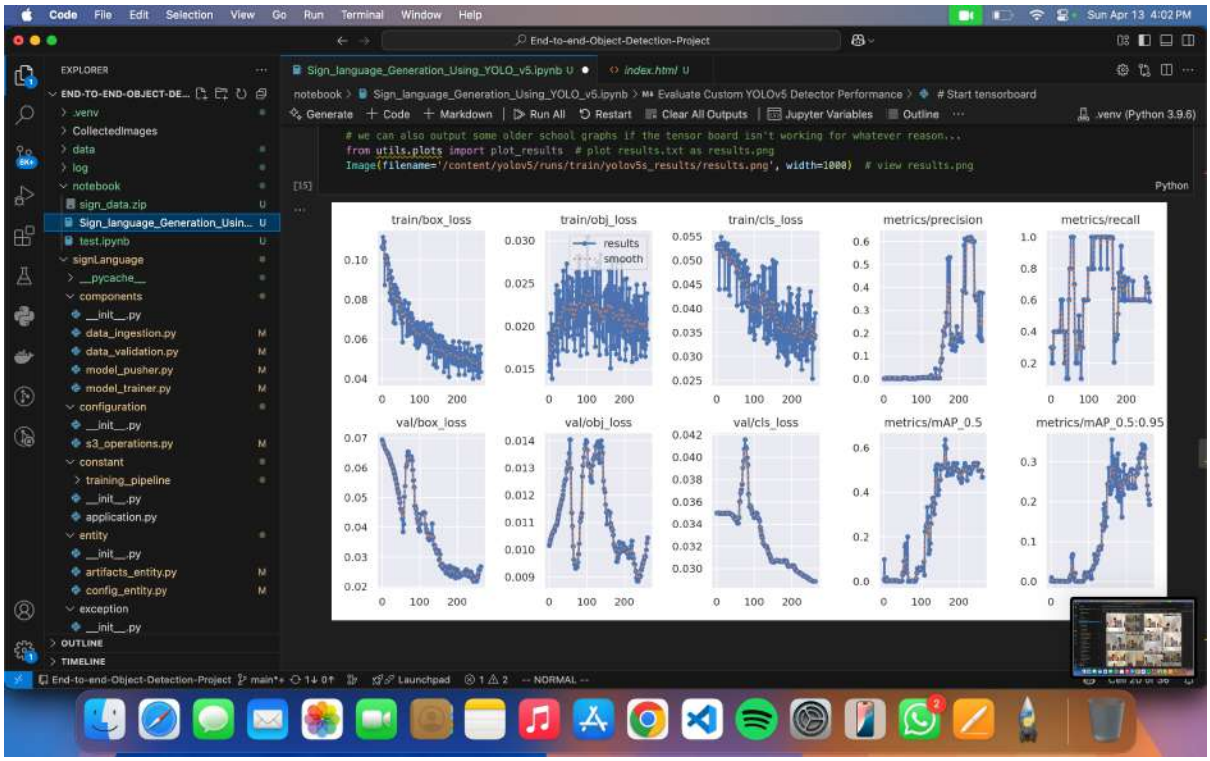


4. After that I have to open any browser and go to localhost:8080/live. Which enables the camera to read the data live.
5. I have manually trained model using my images to perform the prediction.
6. Here is the samples of the data I have collected



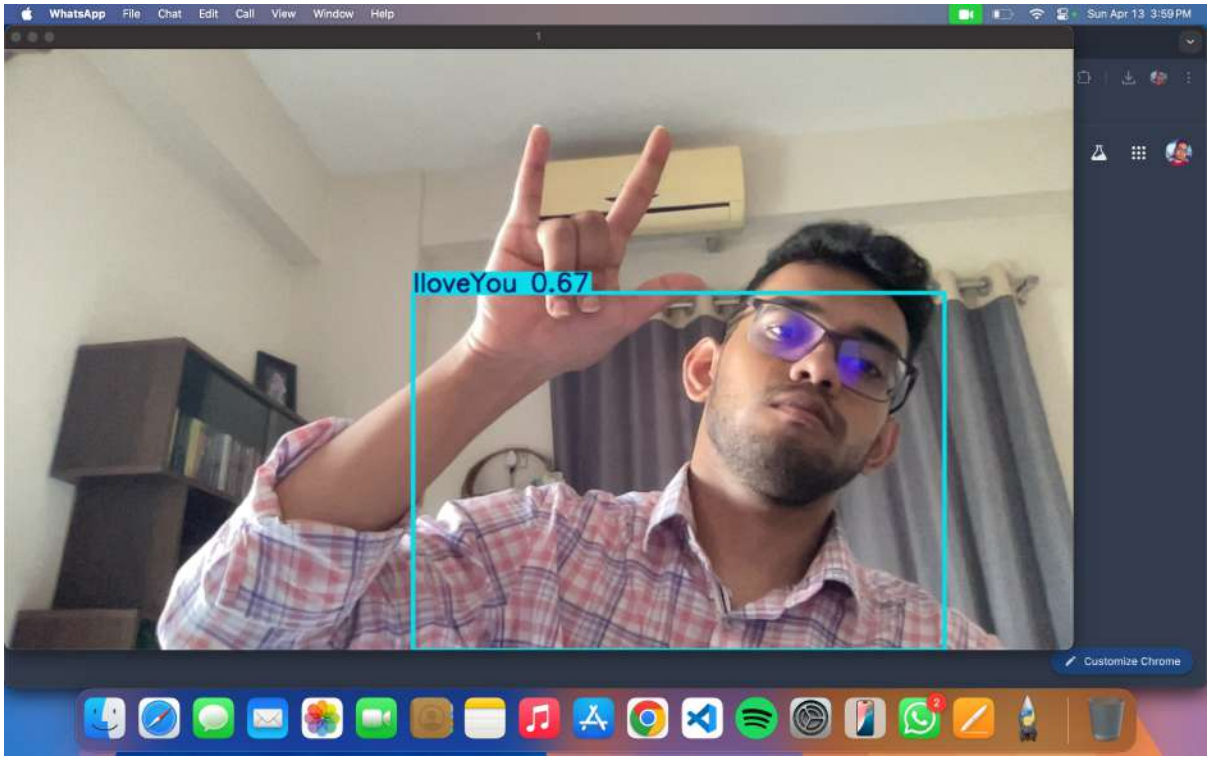


7. I have used tensor flow to determine the metrics of prediction



Results:

The project can detect the sign language part easily with around 67 percent accuracy. The data base needs to be more polished. Rather than that everything is working fine. The image of the working demo is attached below



Conclusion:

The project establishes important progress in accessibility by creating a sign language detection system through object detection techniques. The project achieves accurate and efficient hand gesture interpretation through YOLOv5 combined with PyTorch and supportive tools that include Roboflow and Conda.

The developed system creates fundamental structures for assistive technology applications which provide realtime sign language translation into speech or text so hearing and non-hearing individuals can communicate with each other. The project emphasizes its focus on technical innovation as well as social impact achievements by concentrating on data collection and algorithm development and system assessment.

Future development of the project will involve bigger data resources and sentence-level processing capabilities and mobile application integration. This effort allows the development of accessible technological systems that support all users through their communication methods.