SIGN LANGUAGE RECOGNITION USING MACHINE LEARNING

A PROJECT REPORT

Submitted by

YOGESH I B	**[211420104313]**
HARRESH A	**[211420104094]**
SOMPALLI VENU	**[211420104257]**

in partial fulfillment for the award of the degree
of

BACHELOR OF ENGINEERING
in

COMPUTER SCIENCE AND ENGINEERING

PANIMALAR ENGINEERING COLLEGE
(An Autonomous Institution, Affiliated to Anna University, Chennai)

APRIL 2024

PANIMALAR ENGINEERING COLLEGE

(An Autonomous Institution, Affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"SIGN LANGUAGE RECOGNITION USING MACHINE LEARNING"** is the Bonafide work of **"YOGESH I B (211420104313), HARRESH A(211420104094), SOMPALLI VENU(211420104257)"** who carried out the project work under my supervision.

Signature of the HOD with date

Dr L JABASHEELA M.E., Ph.D.,

Professor and Head,

Department of Computer Science and Engineering,

Panimalar Engineering College,

Chennai – 123

Signature of the Supervisor with date

Dr. V Sathya Preiya, M. C. A.,
M. Phil., M. E., P.h.D.,

Associate Professor,

Department of Computer Science and Engineering,

Panimalar Engineering College,

Chennai- 123

Submitted for the Project Viva – Voce examination held on ____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

DECLARATION BY THE STUDENT

We **YOGESH I B(211420104313), HARRESH A(211420104094) & SOMPALLI VENU(211420104257)** hereby declare that this project report titled **“SIGN LANGUAGE RECOGNITION USING MACHINE LEARNING”** , under the guidance of **Dr. V Sathya Preiya,M. C. A, M. Phil., M. E., Ph.D & Dr.N.Pughazendi,M.E,Ph.D** is the original work done by us and we have not plagiarized or submitted to any other degree in any university by us.

YOGESH I B (211420104313)

HARRESH A(211420104094)

SOMPALLI VENU (211420104257)

ACKNOWLEDGEMENT

Our profound gratitude is directed towards our esteemed Secretary and Correspondent, **Dr. P. CHINNADURAI, M.A., Ph.D**.,for his benevolent words and fervent encouragement. His inspirational support proved instrumental in galvanizing our efforts, ultimately contributing significantly to the successful completion of this project

We want to express our deep gratitude to our Directors, **Tmt. C. VIJAYARAJESWARI, Dr. C. SAKTHI KUMAR, M.E., Ph.D., and Dr. SARANYASREE SAKTHI KUMAR, B.E., M.B.A., Ph.D.,** for graciously affording us the essential resources and facilities for undertaking of this project.

Our gratitude is also extended to our Principal, **Dr. K. MANI, M.E., Ph.D.,** whose facilitation proved pivotal in the successful completion of this project.

We express my heartfelt thanks to **Dr. L. JABASHEELA,M.E., Ph.D.,** Head of the Department of Computer Science and Engineering, for granting the necessary facilities that contributed to the timely and successful completion of project.

We would like to express our sincere thanks to **Project Coordinator Dr N PUGHAZENDI M.E., Ph.D.,** and **Project Guide Dr. V Sathya Preiya, M. C. A., M. Phil., M. E., Ph.D.,** and all the faculty members of the Department of CSE for their unwavering support for the successful completion of the project.

YOGESH I B (211420104313)

HARRESH A(211420104094)

SOMPALLI VENU (211420104257)

ABSTRACT

Sign language is an important means of communication for deaf people, facilitating interaction with the wider community. This project presents a comprehensive sign language recognition system designed to improve communication accessibility. The system includes Convolutional Neural Networks (CNN) for landmark detection, hybrid CNN-Recurrent Neural Networks (RNN) for motion recognition, and modular architecture for scalability and flexibility. The project begins with a reliable data collection module in collaboration with individuals who know sign language to compile a diverse database. Landmark Detection, a key step, uses CNN to accurately identify hand landmarks and capture the nuances of hand gestures. Finally, the Preprocessing module ensures optimal data quality, and the Recognition module combines CNNs and RNNs to interpret signals in real time. The video captures module interfaces seamlessly with standard webcams, making the system accessible for real-world applications. The user interface module provides visual feedback on recognized actions, adding an interactive and user-friendly experience. Continuous learning is facilitated through feedback and improvement modules, collecting user feedback and regularly refining the system. The project demonstrated the effectiveness of the sign language recognition system through a thorough evaluation, including metrics such as accuracy, precision, and recall. Future ideas include online learning algorithms for real-time adaptation and additional features to increase the inclusiveness of the system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF ABBREVIATIONS

RNN	RECURRENT NEURAL NETWORK
CNN	CONVOLUTIONAL NEURAL NETWORK
ASL	AMERICAN SIGN LANGUAGE
SLR	SIGN LANGUAGE RECOGNITION
GSL	GREEK SIGN LANGUAGE
OCR	OPTICAL CHARACTER RECOGNITION
NLP	NATURAL LANGUAGE PROCESSING
UEFI	UNIFIED EXTENSIBLE FIRMWARE INTERFACE
RGB	RED, GREEN, BLUE
LSTM	LONG SHORT-TERM MEMORY
TP	TRUE POSITIVE
FP	FALSE POSITIVE
TN	TRUE NEGATIVE
FN	FALSE NEGATIVE

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Language recognition using machine learning is a revolutionary application that solves communication problems for deaf and hard of hearing people. The system uses computer vision and neural networks to interpret hand movements and convert them into text and speech. It involves training different samples of data to recognize the subtle changes and configurations that make up the narrative. We can integrate this project in other applications to make more applications accessible for the deaf and dumb people. Challenges include adapting to changes in sign language, overcoming background noise, and improving accuracy in different environments. The potential impact of this technology on education, work and social relationships, promoting greater independence and participation for the hearing impaired. Ongoing research focuses on developing accurate algorithms, addressing cultural and regional differences in sign language, and improving user interaction. As technology continues to advance, the integration of machine learning and artificial intelligence promises to create a more accessible and inclusive environment, disrupting communication and enabling deaf people to integrate into a world mostly designed for communication.

1.2 OBJECTIVES

- *Developing a reliable sign language recognition system*

Design and implement a system that can accurately recognize gestures in different languages while ensuring reliability in different scenarios around the world.

- *Improve communication accessibility*

Improve accessibility for people with disabilities by creating an interface that allows themto communicate consistently through the user's language.

- *Use an advanced neural system to identify the lander*

Use a convolutional neural network (CNN) to capture important spatial information for accurate landmark detection and hand gesture recognition.

- *Combine temporal dynamics with a hybrid CNN-RNN model*

Apply a hybrid model combining CNNs and recurrent neural networks (RNNs) to efficiently capture spatial and temporal features in sign language behavior.

- *Ensure scalability and modularity*

Designing an extensible system architecture that allows for future improvements andmodularity that facilitates easy integration of additional features or improvements.

- *Collect and organize different sign language datasets*

Work with individuals who know sign language to develop a comprehensive informationset that includes multiple instructions and scenarios for effective learning.

1.3 PROBLEM DEFINITION

Sign language recognition aims to develop machine learning systems capable of accurately interpreting and understanding gestures and signs made by individuals using sign language. The problem involves the translation of visual signals captured through images or videos of signers into corresponding textual or symbolic representations of the conveyed message. This entails recognizing the intricate movements and configurations of hands, fingers, and other body parts involved in forming signs, as well as accounting for variations in sign language dialects and styles across different regions and communities. Sign language recognition systems must address challenges such as variability in sign appearance, background clutter, occlusion, and the dynamic nature of signing gestures. The ultimate goal is to enable effective communication between sign language users and non-signers, facilitating accessibility and inclusivity in various domains, including education, communication, and technology.

CHAPTER 2

LITERATURE REVIEW

P. Kumar, S. Swetha and M. Sundari(2021), offered a sign language recognition system that translates sign language signals into written words. The system uses a webcam for video recording, OpenCV for frame capture, Holistic Media Pipe for feature extraction, and an LSTM network for gesture recognition and translation. The goal of the system is to improve communication with the deaf. It discusses various aspects of the system, including database creation, preprocessing, feature extraction, classification algorithms, and the use of deep learning techniques. It also discusses challenges that arise during development, such as database creation and real-timeperformance. This concludes with suggestions for future work to expand the database andimprove accuracy using deep learning techniques.[1]

M. Madhiarasan Partha Pratim Roy(2022), It provides a comprehensive overview of language recognition, discussing different sign languages, methods, and databases. Complex backgrounds, lighting limitations, and limited databases present challenges and challenges related to sign language recognition. It also provides an overview of state-of-the-art developments and models in the field, including various feature extraction techniques and classification architectures. It identifies research gaps and limitations and suggests future directions for sign language recognition, including developing user-friendly and robust models, improving database quality, and exploring multimodal approaches. In general, the aim is to attract the attention and understanding of linguistic researchers.[2]

Sharvani Srivastava, Amisha Gangwar, Richa Mishra, Sudhakar Singh(2021), It discusses the development of a real-time sign language recognition (SLR) system usingthe TensorFlow object detection API. The system is trained on a database of Indian character alphabets captured by Webcam. Transfer learning was used to train the model with an average confidence level of 85.45%. The system can detect signs in sign languageand translate them into English. It also addresses challenges in SLR research such as the limitations of large databases and misconceptions about sign language. Future directions include expanding the database and exploring different models for different symptoms.[3]

Deep Rameshbhai Kothadiya(2022), Presenting a methodology for Indian character recognition using deep learning algorithms, specifically LSTM and GRU. The model does not require any special environment or camera settings to be suitable for real-world scenarios. This technique involves dividing the video file into separate sub-videos containing different words, extracting frames from these videos and using InceptionResNetV2 to extract features from the frames. These features are then fed to a recurrent neural network (RNN), which is a combination of LSTMand GRU, to predict the correct word. The model achieves high accuracy in recognizing signal signals.[4]

Reddygari Sandhya Rani , R Rumana , R. Prema(2021) , Discusses the development of a real-time vision-based system for American Sign Language (ASL) recognition. This system aims to recognize hand gestures used in sign language for effective communication with deaf and dumb people. It uses a single cameraand relies on certain assumptions such as the user within a defined perimeter area and distance. The system faces challenges related to database generation, filter selection, and model accuracy. Various methods and techniques such as Discrete Wavelet Transform (DWT) and Principal Component Analysis (PCA) have been studied to recognize hand gestures. The system achieved 92% accuracy in the database. Overall, it demonstrates theimportance of sign language recognition systems in facilitating communication for peoplewith hearing or speech impairments.[5]

Suharjito,Ricky Anderson(2017), The Sign Language Recognition (SLR) system has been developed by researchers using various data acquisition methods, including cameras, Microsoft Kinect, sensory gloves, and accelerometers. Researchers have also created their own datasets for training the SLRsystems, as there is a lack of available datasets for different sign languages. The classification methods used in SLR include Hidden Markov Models (HMM), Convolutional Neural Networks (CNN), Support Vector Machines (SVM), Self- Organizing Maps (SOM), and Multilayer Perceptron (MLP). The accuracy of the SLR systems varies depending on the method used, with the highest accuracy achieved by the 3D-CNN method at 94.2%[6]

Ananthajothi K, Karthick T, Amanullah M(2022), The paper describes the design of a sign language recognition system using an intelligent glove. The glove captures hand gestures and translates them into speech or understandablelanguage. The system collects data from 100 individuals wearing the gloves and stores it in a CSV format. The data is then preprocessed and normalized using PCA for feature extraction. The system uses flex sensors, an inertial measurement unit (IMU), and a Hall sensor to recognize hand movements. Three different machine learning algorithms (SVM, Naïve Bayes, and Decision Tree) are implemented for classification, with SVM achievingthe highest accuracy of 90%. The system also includes a speech converting function and a speaker for audible results. The data is collected from the sensors and sent to a Firebase database, which is then received by a Raspberry Pi device for translation and speech output. The paper also mentions the use of a mic and a USB connector for voice input and text conversion, as well as the use of a camera or other sensor for capturing facial expressionsin future work. The proposed system aims to bridge the communication gap for speech- disabled individuals and improve language learning deficits.[7]

K Ashokkumar, S Parthasarathy, S Nandhini, K Ananthajothi(2022), It discusses the development of a Sign Language Recognition (SLR) system using Convolutional Neural Networks (CNNs) to bridge the communication gap between normal people and deaf people. The system aims to recognize Indian Sign Language (ISL) gesturesand translate them into text or speech. The process involves capturing images of hand gestures using a web camera, preprocessing the images using computer vision techniques,training the CNN model to identify the gestures, and predicting the name of the captured image. The model achieved an accuracy of about 95%. The authors propose further enhancements to enable bidirectional translation between sign language and spoken language.[8]

Dhirendra Kumar Choudhary(2021) , It discusses the challenges and potential solutions in designing a gesture recognition system for Sign Language. It specifically focuses on three main topics: the simultaneity of information conveyed by manual signs, the synchronicity between the two hands, and the different classes of signs encountered in a Sign Language sentence. It also mentions that Sign Language is a full-featured language with lexicons and grammar rules, and that sentences in Sign Language involve not only hand gestures but also face mimics, gaze, and torso movements. It aims to address the recognition and synchronization problems that arise in the recognition of manual signs, particularly in relation to the simultaneous use ofthe two hands. It describes the different types of signs and their features, as well as the interactions and relationships between the hands. The paper concludes by proposing an architecture to solve the synchronization problems.[9]

Vikas G N(2021), It discusses the development of a sign language detection system using action recognition. The system aims to enable hearing-impaired individuals to benefit from deep learning advancements. It utilizes video analysis and gesture recognition to detect and interpret signlanguage actions performed by humans. The system employs contour analysis, feature extraction, and LSTM layers to build an effective communication model. The software can detect sign language in real-time, facilitating communication between muted and normal individuals. It also mentions the use of transfer learning to improve learning speed and classification accuracy. Overall, the system aims to provide accessibility and improve communication for the hearing-impaired community.[10]

Dipalee Golekar(2022), It discusses the development of a sign language detection system using action recognition. The system aims to enable hearing-impaired individuals to communicate more easily by analyzing and responding to sign language gestures. The system utilizes video footage andemploys contour analysis and feature extraction techniques to detect sign language and itsvariations. [11]

Sabari Priya A(2023), It discuses the development of a sign language recognition system using image processing techniques. The system aims to bridge the communication gap between individuals with speaking and hearing abilities and those without. It utilizes a user-hand gesture-only human-computer interface (HCI) and convolutional neural networks (CNN) for hand gesture recognition. The system converts hand gestures captured by a camera into textual descriptions and then converts them into speech for the user. The proposed system includes recognition of numbers, alphabets, and common words in American Sign Language (ASL). The system's architecture involves creating a database of training images, extracting features using CNN, and training a classifier for classification. Testing and validation are performed using k-fold cross-validation. The system utilizes technologies such as TensorFlow, OpenCV, Keras, and NumPy. The hardware requirements include a camera, sufficient RAM and GPU, and a monitor and keyboard.

CHAPTER 3

THEORITICAL BACKGROUND

3.1 IMPLEMENTATION ENVIRONMENT

3.1.1 Dataset Description

The input module consists of 300 instances and 6 Indian Sign Language words and landmarks for each instance. This dataset was meticulously created be augmenting existing datasets on Indian Sign Language and recording that through webcam. Then we labelled the dataset collected for further use.

3.1.2 Performance Evaluation etrics

***Accuracy*:** It measures the percentage of correctly predicted sign language actions acrossdatabase.

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

True Positive (TP)

Meaning: The model assumes that sign language behavior is positive (recognized ascorrect).

Example: The model correctly identifies the letter "A" as "A".

False positive (FP)

Meaning: The model incorrectly predicts sign language behavior when it is negative(misrecognized).

Example: The model incorrectly identifies an action as the letter "B" when it is the letter"A".

True Negative (TN)

Meaning: The model predicts sign language behavior as negative (true rejection of false recognition).

Example: The correct model determines that non-symbolic actions are not part of thesymbolic alphabet.

False Negative (FN):

Meaning: The model assumes that sign language behavior is negative when it is positive(cannot recognize the correct sign).

Example: The model does not recognize the letter "A" and thinks it is not a symbol. We achieved a accuracy of 92% using hybrid of CNN and RNN in this project

3.2 SYSTEM ARCHITECTURE

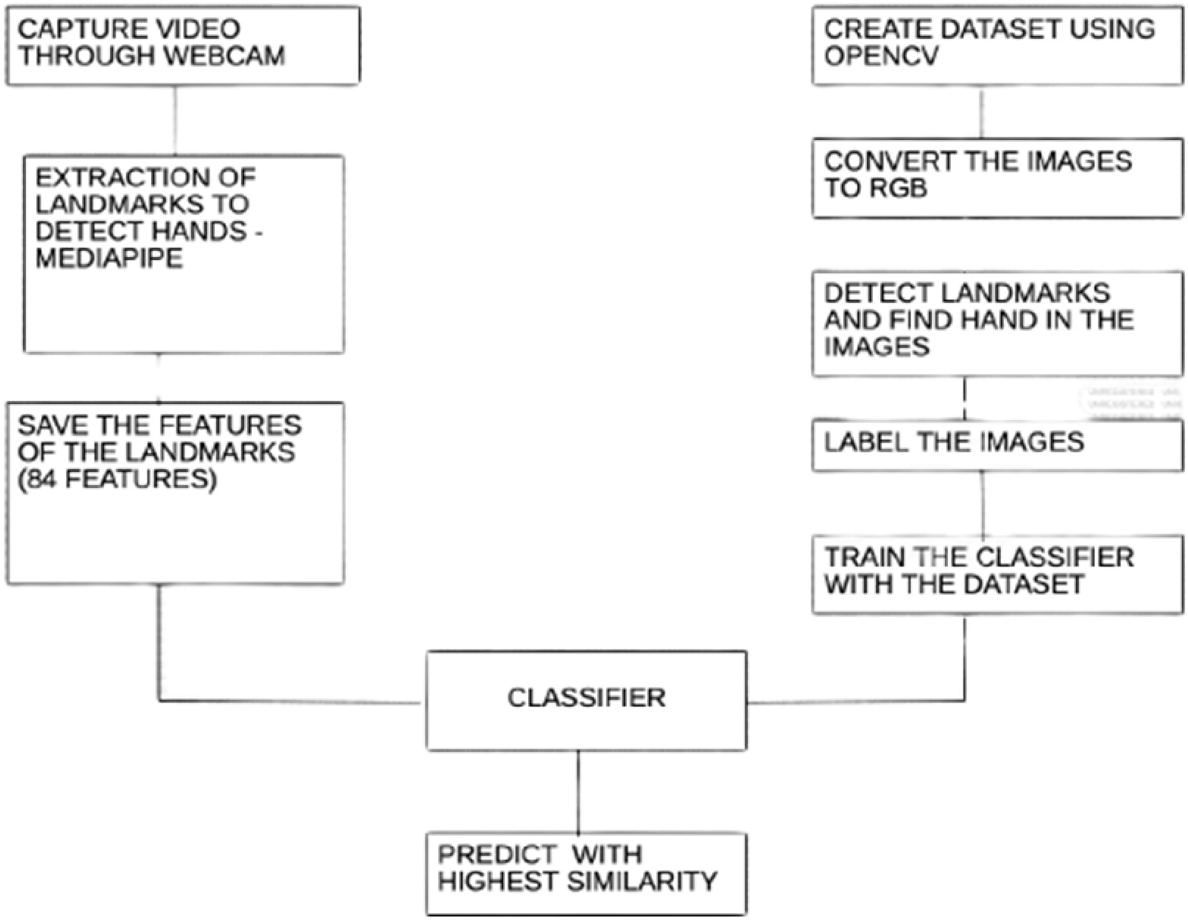


Fig 3.2.1 System Architecture

The above system architecture diagram Fig.3.2.1 deals with the flow from data collection tothe performance evaluation of the mode. The input from the user and the preprocessing ofthe Sign language dataset and it to look for patterns is the first step in the design model. After recognizing the signs it is given as text to the user. A typical sign language recognition system utilizes machine learning to translate hand gestures into recognizable signs. The process begins with capturing a live video stream of the user signing using a webcam. To identify the hands within the video frames, Media Pipe, an open-source framework, extracts landmarks. These landmarks pinpoint specific locations on the hands, like fingertips and wrists. Following this, the system converts the video frames into a standard format suitable for the classifier and refines the hand location within the image. From the identified landmarks, a set of features is extracted, which could be distances between fingertips or joint angles. These features create a unique representation of the hand posture. This data is then labelled with the corresponding sign it represents to establish a training dataset for the machine learning model. Finally, a machine learning classifier, often a convolutional neural network, is trained using this labelled dataset. Once trained, the system can predict the sign from a new video frame by comparing its extracted features to those in the training data and identifying the sign with the most similar features.

3.3 PROPOSED METHODOLOGY

The proposed sign language recognition system is designed to be a comprehensive, user- friendly and adaptive platform that meets the communication needs of the deaf. The systemincludes several key features and functions aimed at high accuracy, real-time performanceand continuous improvement.

- **Land Mark Detection with CNNs**

Using Convolutional Neural Networks (CNN), the proposed system will make accurate annotations on hand gestures and obtain important spatial information for accurate recognition. This feature ensures that the system can interpret subtle nuances in the signal.

- **Hybrid CNN-RNN model for gesture recognition**

The core of the recognition module will include a hybrid model that combines Convolutional Neural Networks (CNN) to extract spatial features with Recurrent Neural Networks (RNN) to capture temporal dynamics. This allows the system to interpret the static and dynamic aspects of sign language gestures.

- **Show Real Time Inference**

The system will provide real-time recognition and feedback, allowing users to communicate seamlessly and receive immediate responses. Achieve low-latency processing, real-time search feature improves user experience and facilitates naturalcommunication.

- **User Interface for Visual Feedback**

A user-friendly interface will be created to provide visual feedback on actions taken. It includes recognized sign language symbols, corresponding text labels, and instructional guidelines to guide users in effective communication.

The following advantage are

- **Safe handling of dynamic signals:**

Advanced algorithms allow for accurate recognition of dynamic gestures, capturing temporal dynamics with precision.

- **Actual processing efficiency:**

The system has real-time functionality, improving user experience in instant response and interactive applications.

3.4 MODULE DESIGN

3.4.1 System Design

The system design for this project emphasizes modularity for seamless integration, incorporating advanced machine learning algorithms for precise sign language recognition. Real-time processing capabilities enable swift gesture interpretation, while intuitive interfaces facilitate user interaction and feedback collection. Flexibility is prioritized to accommodate future updates and enhancements, ensuring adaptability to evolving needs and technological advancements.

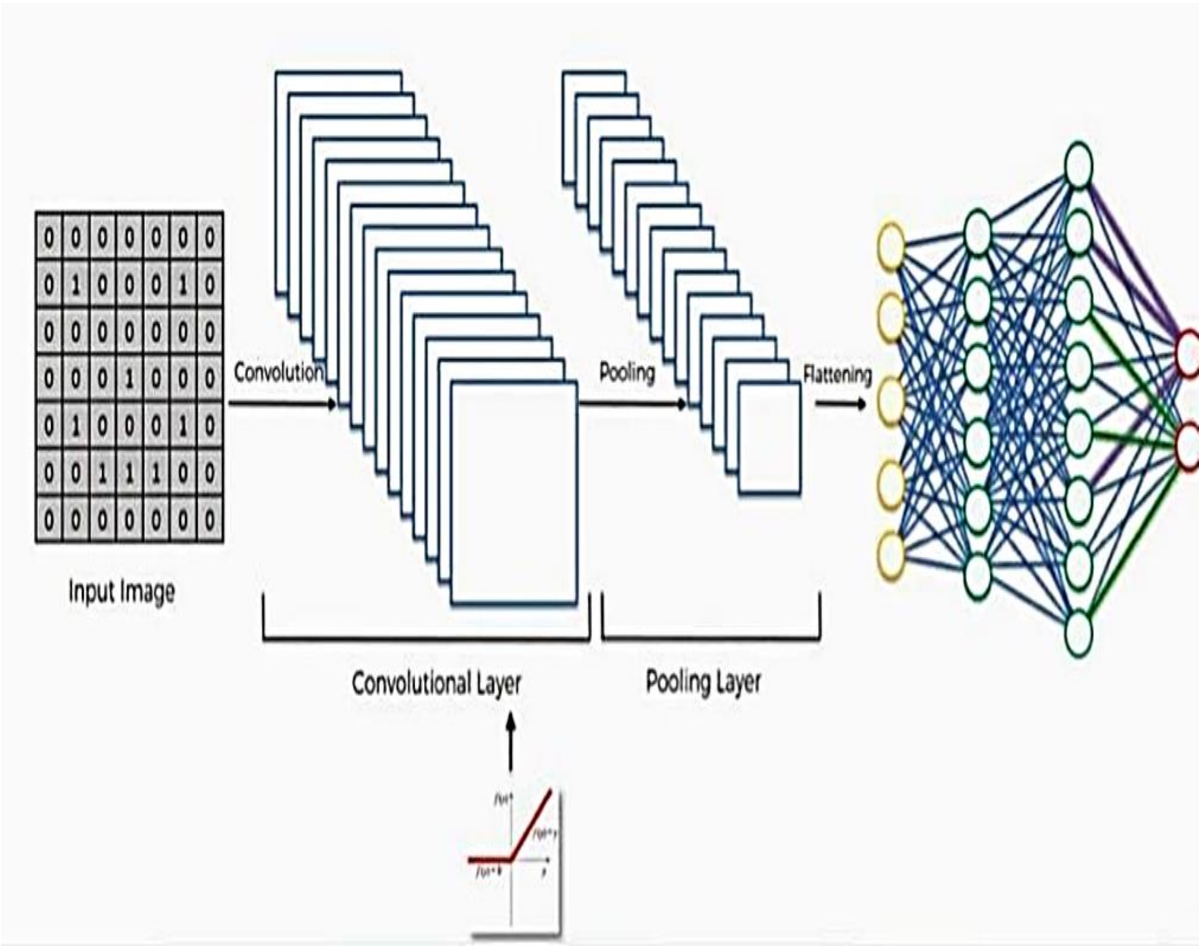


Fig 3.4.1.1 System Design

Data analysis involves inferring, understanding, and providing insight from collected data. Fig 3.4.1.1 shows the steps involved in Data Analysis. Using technologies such as optical character recognition (OCR) and natural language processing (NLP), data is converted into machine-readable text, enabling keyword extraction, opinion analysis and corporate recognition. Patterns and concepts are identified, classification is done, modeling is done, and data quality is ensured. Machine learning-powered automation improves analysis of large amounts of text in various fields such as law, business, and science, making it easier to make quick decisions and gain insight into the extraction process

3.4.2 Sequence Diagram

Sequence diagram displays in Fig 3.4.2.1 the time sequence of the objects participating in the interaction. This consists of the vertical dimension(time) and horizontal dimension (different objects). The sequence diagram shows the interaction of objects presented chronologically. It refers to the order of messages to be exchanged between the objects included in the view and the objects needed to complete the class and visual functionality. Sequence diagrams are commonly used the system under development k is obtained in the case of logical approach. Events like events are events or events that happen. Sequential diagrams show the sequence in which messages are exchanged between parallel vertical lines (lifelines), different processes or objects that exist as a horizontal arrow.It allows the specification of simple runtime scenarios in a graphical way. If the life line belongs to an object, it indicates a role. The name of the instance is Informal and Unmanual Ka Transplant.

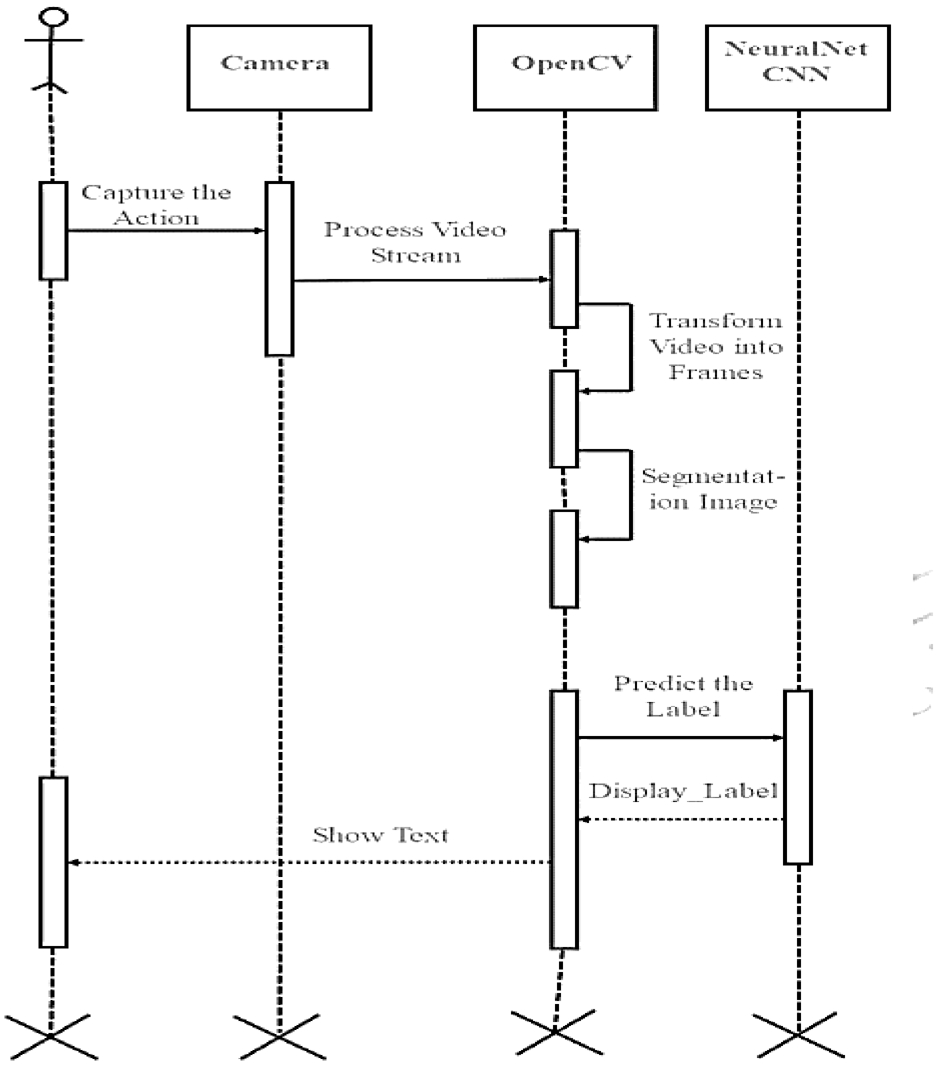


Fig 3.4.2.1 Sequence Diagram

3.4.3 Use Case Diagram

Data is used during requirements elicitation and analysis to represent the functionality of the system. Fig 3.4.3.1 shows the use case of various users using this system. The use case describes the operation of the system that produces visible results for actors. The identification of actors and use cases leads to the definition of boundaries, that is, the difference between activities performed by the system and activities performed by the environment. Actors are outside the system boundaries, information is within the system boundaries. Use information to describe the behavior of the system from the actors' point of view. It defines the functions provided by the system as events that produce visible results for the participants.

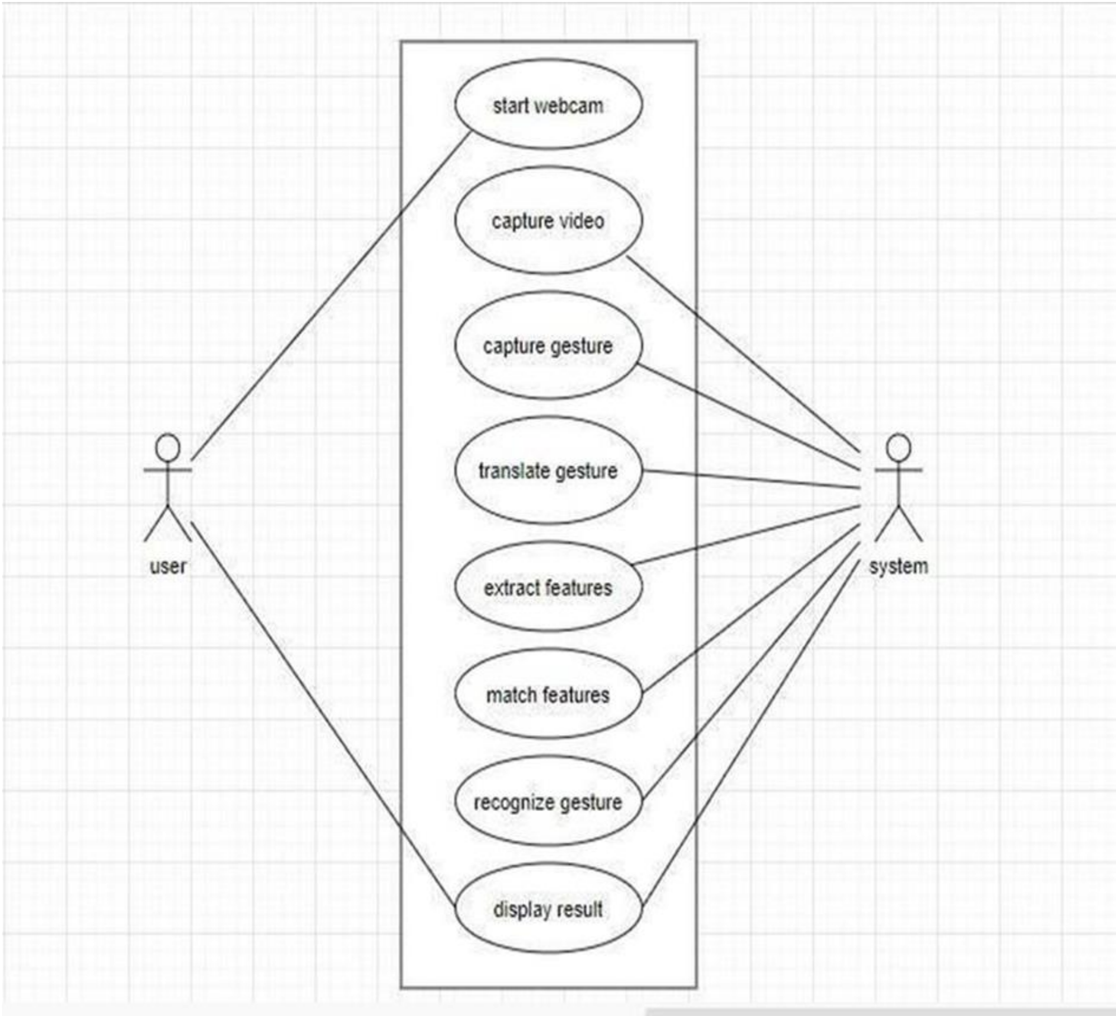


Fig 3.4.3.1 Use Case Diagram

3.4.4 Activity Diagram

Sign language recognition using machine learning follows a step-by-step process. In Fig 3.4.4.1 the system captures a sign language gesture via camera. This image or video data then undergoes preprocessing, where it's cleaned and standardized for better analysis. Next, key features like hand position and finger shapes are extracted. These features are fed into a trained machine learning model, which then classifies them and recognizes the corresponding sign language symbol. Finally, the system outputs the recognized sign in a user-friendly format, such as text or speech.

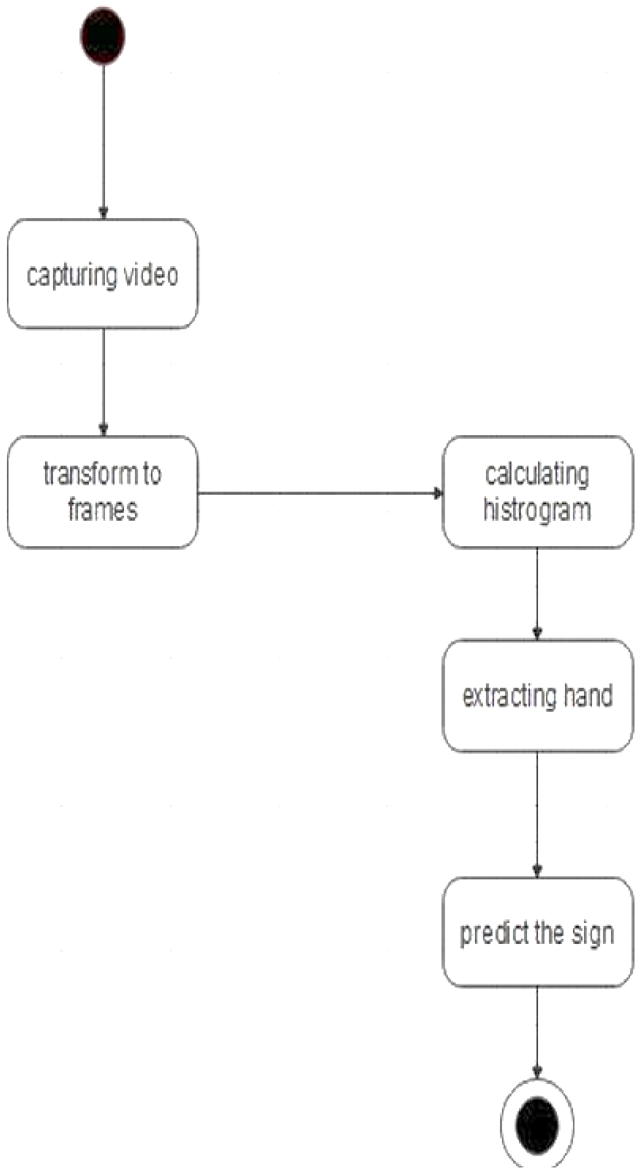


Fig 3.4.4.1 Activity diagram

3.4.5 Collaboration Diagram

A collaboration diagram in a project report illustrates the interactions between different components, modules, or entities within the system. It showcases how various elements collaborate to achieve the overall functionality of the project. Typically, it depicts the flow of communication, data exchange, and dependencies between different parts of the system. In essence, the collaboration diagram provides a visual representation of the system's architecture and highlights the relationships and interactions among its components. This diagram serves as a valuable tool for understanding the system's design and implementation, facilitating communication among project stakeholders, and guiding further development and refinement efforts.

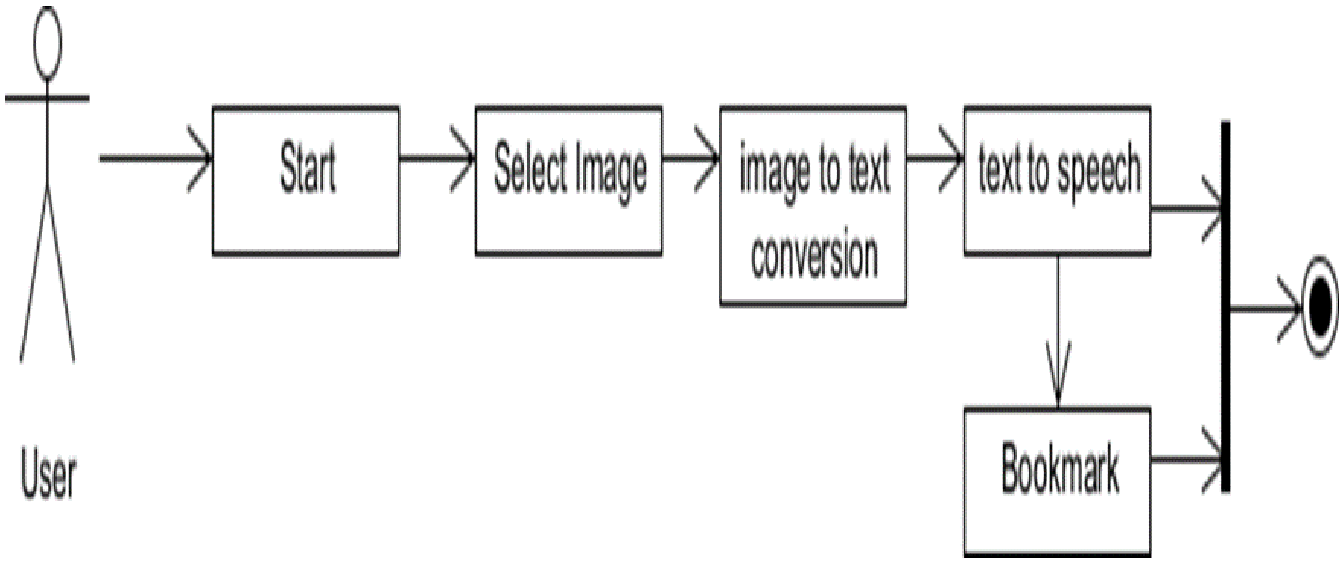


Fig. 3.4.5.1 Collaboration Diagram

Data analysis involves inferring, understanding, and providing insight from collected data. Fig 3.4.5.1 shows the steps involved in Data Analysis. Using technologies such as optical character recognition (OCR) and natural language processing (NLP), data is converted into machine-readable text, enabling keyword extraction, opinion analysis and corporate recognition. Patterns and concepts are identified, classification is done, modeling is done, and data quality is ensured. Machine learning-powered automation improves analysis of large amounts of text in various fields such as law, business, and science, making it easier to make quick decisions and gain insight into the extraction process. In addition to depicting communication flows, collaboration diagrams often include annotations detailing the nature of interactions, such as method calls, messages, or data exchanges. They help identify dependencies between components, aiding in the identification of potential bottlenecks or points of failure. Collaboration diagrams promote clarity and transparency in system design, fostering effective collaboration among team members and stakeholders. By visualizing the system's structure and behavior, they facilitate comprehensive understanding and informed decision-making throughout the project lifecycle.

3.4.6 Class Diagram

A class diagram in a project report illustrates the structure of the system by representing classes, attributes, methods, and their relationships. It provides a visual blueprint of the system's object-oriented design, showcasing the entities and their interactions. Class diagrams help in understanding the system's architecture, facilitating code implementation and maintenance. They depict inheritance, association, aggregation, and composition relationships between classes, aiding in the identification of key components and their functionalities. Through concise representation, class diagrams serve as essential documentation for developers and stakeholders, enhancing collaboration and communication throughout the project.

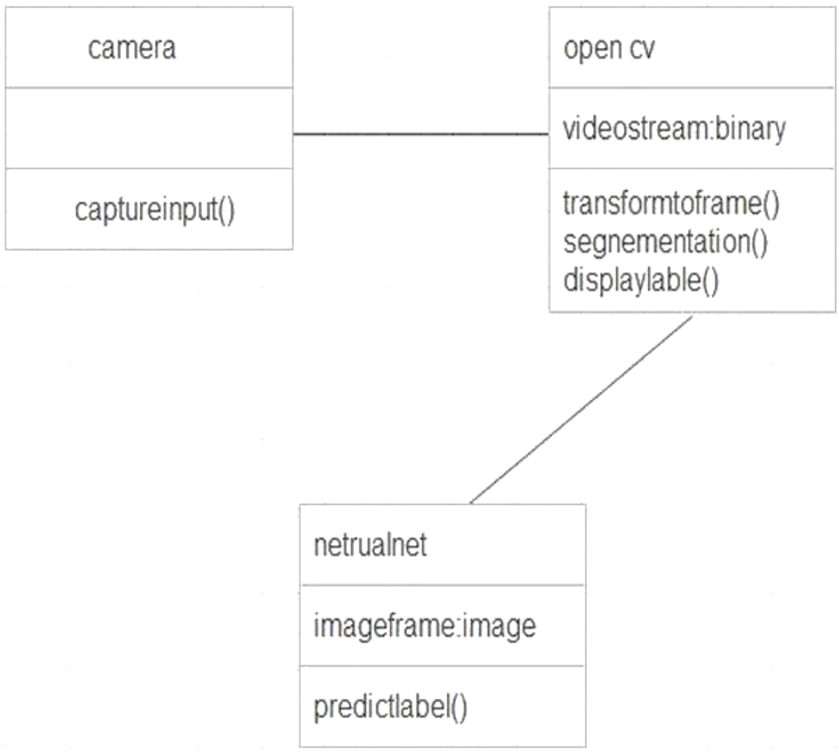


Fig 3.4.6.1 Class diagram

Class diagrams model class structure and contents using design elements such as classes, packages and objects. Fig 3.4.6.1 describe the different perspective when designing a system-conceptual, specification and implementation. Classes are composed of three things: name, attributes, and operations. Class diagram also display relationships such as containment, inheritance, association etc. The association relationship is most common relationship in a class diagram. The association shows the relationship between instances of classes

3.4.7 Deployment Diagram

A deployment diagram in a project report visually depicts the physical architecture of the system, showcasing hardware components and their interconnections. It illustrates how software artifacts are distributed across different nodes, servers, or devices. Deployment diagrams highlight deployment configurations, including servers, databases, and networking infrastructure. They provide insights into system scalability, redundancy, and fault tolerance strategies. By mapping software components to physical resources, deployment diagrams facilitate efficient system deployment, management, and maintenance.

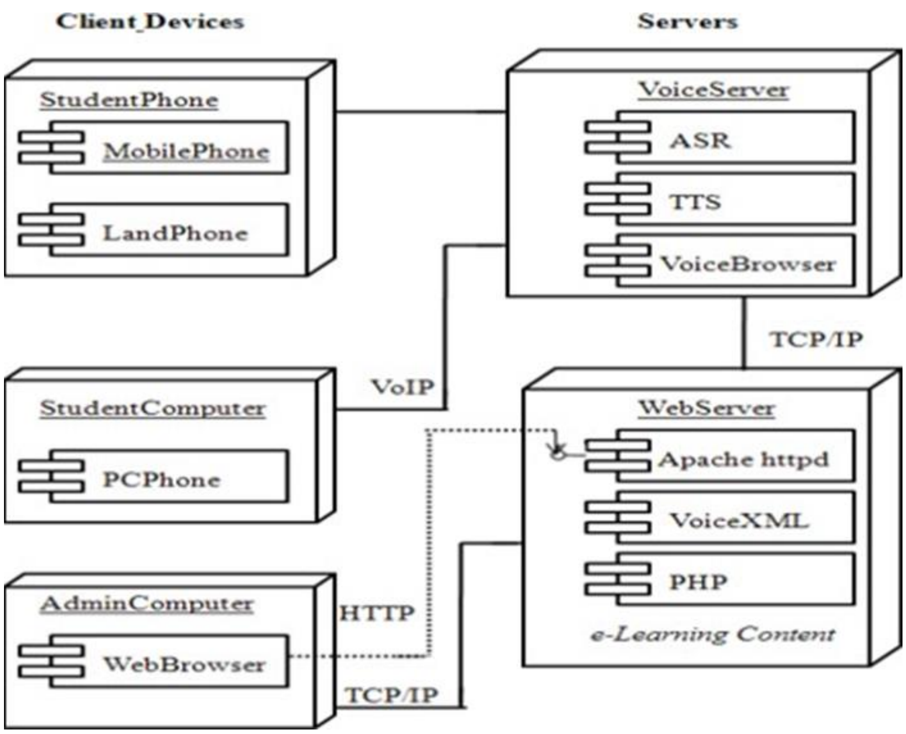


Fig3.4.7.1 Deployment Diagram

A deployment diagram Fig 3.4.7.1 is a type of diagram in the Unified Modeling Language (UML) that shows the physical aspects of an object-oriented software system. It models the run- time configuration in a static view, and visualizes the distribution of components in an application. Deployment diagrams are important for visualizing, specifying, and documenting embedded, client/server, and distributed systems. Deployment diagrams also illustrate communication paths between nodes, indicating data flow and network connections. They assist in identifying potential points of failure and optimizing system performance. Deployment diagrams aid in resource allocation and capacity planning, ensuring the system meets operational requirements. By providing a clear overview of the deployment environment, they support effective decision-making and communication among project stakeholders.

3.4.8 Component Diagram

A component diagram in a project report visually represents the high-level structure of the system by decomposing it into cohesive, modular components. Fig 3.4.8.1 illustrates the relationships and dependencies between these components, including interfaces and dependencies. Component diagrams highlight the encapsulation and abstraction of functionality within each component, promoting maintainability and reusability. They aid in understanding the system's architecture and facilitate component-level design discussions. Component diagrams also assist in identifying potential areas for component reuse or replacement, fostering scalability and flexibility in the system. Through concise representation, they serve as valuable documentation for developers and stakeholders, guiding the implementation and evolution of the system.

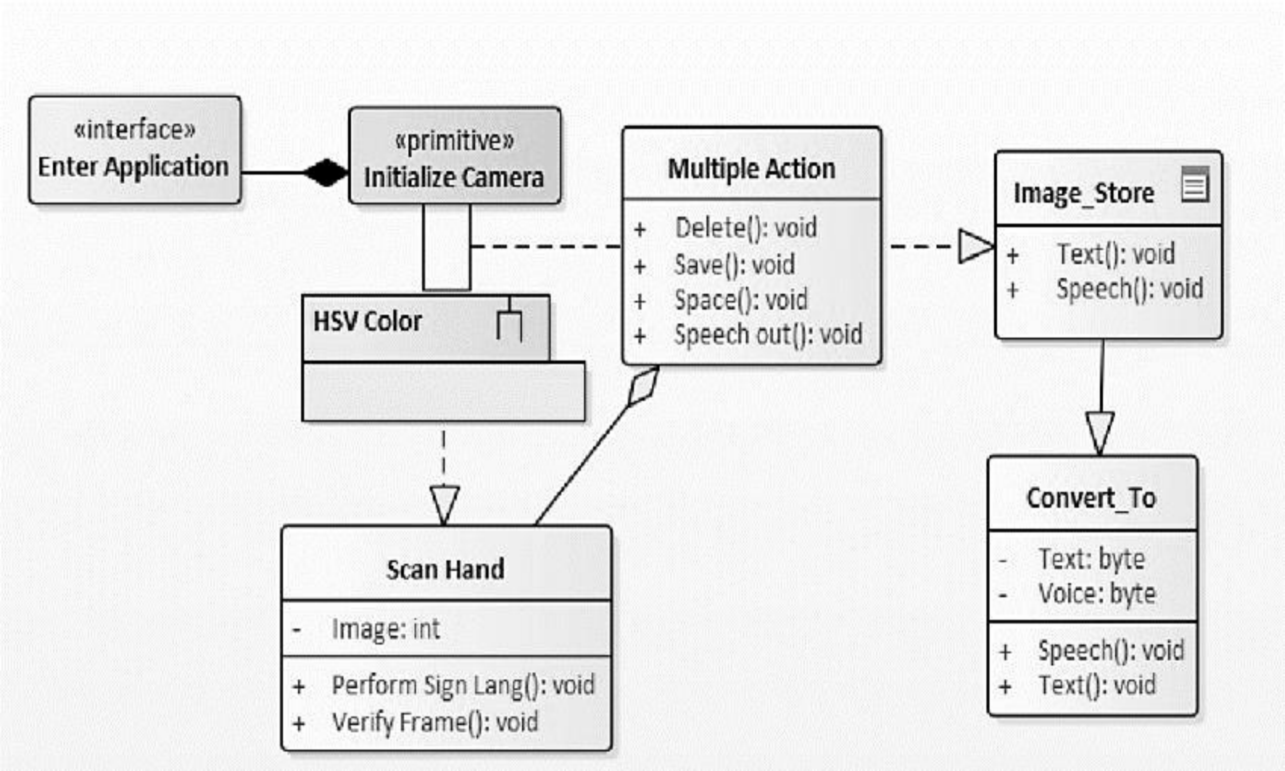
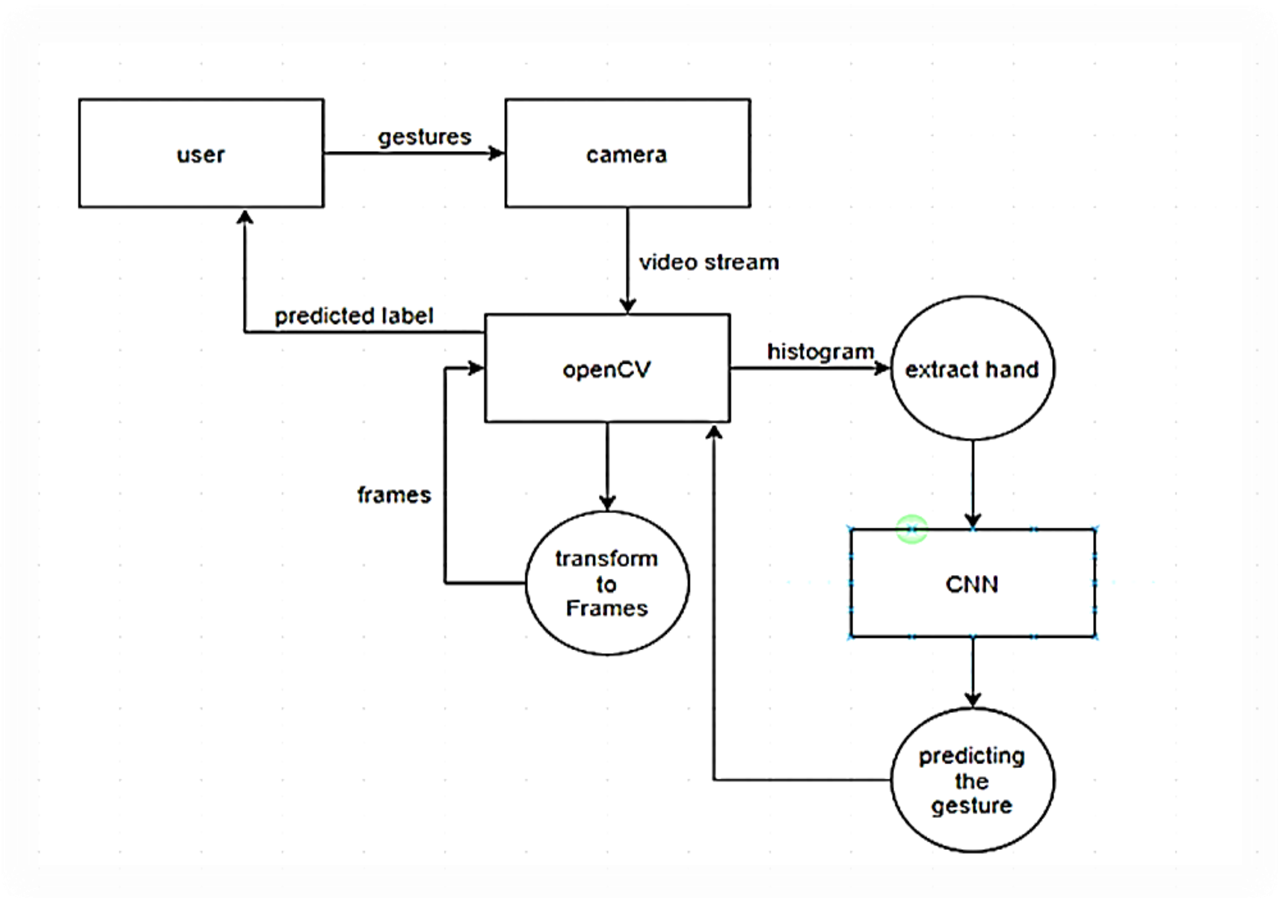


Fig 3.4.8.1 Component diagram

1. The user interacts with the application through an interface.
2. The application captures video or images through a camera.
3. The system performs actions on the captured data, possibly including saving or converting it.
4. An image processing module might perform tasks like hand segmentation.
5. Finally, a module translates signs into text or speech for user output.

3.4.9 Data Flow Diagram

A data flow diagram in a project report visually illustrates the flow of data within the system, depicting inputs, processes, and outputs. It showcases how data moves through different components and modules, highlighting transformations and interactions. Data flow diagrams aid in understanding the system's data architecture and the flow of information between various entities. They help identify data sources, sinks, and storage locations, facilitating data management and integrity. Through clear visualization, they provide insights into system functionality, supporting effective communication and decision-making among project stakeholders.



3.4.9.1 Data Flow diagram

A simple graphical form that can be used to represent data input into the system, various operations performed on the data, and data output from the system. It describes the flowof information in a process or system and how information is processed as input and output. It uses defined symbols, such as rectangles, circles, and arrows, to display datainputs, outputs, storage points, and paths at each location. Fig 3.4.9.1 shows how the data flows at various process in this proposed system.

CHAPTER 4

PROJECT DESCRIPTION

4.1 MODULES

1. Dataset Collection Module
2. Training Module
3. Evaluation Module
4. Video Capture Module
5. Preprocessing Module
6. Recognition Module
7. Feedback and Improvement Module

4.2 MODULE DESCRIPTION

4.2.1 Dataset Collection Module

- *Source of Information*

The dataset used in this project forms an important basis for the development and evaluation of sign language recognition systems. The information includes various sign language gestures, which include various gestures commonly used in communication. The key resource for obtaining this database is collaboration with deaf speakers, including members of the deaf and hearing community.

- *Description of Information*

To facilitate the training of the Landmark Detection Model, each sign language in the database is described in detail with the handprint. Annotators are trained to identify and mark the key points of the hands and pick up the nuances of the gestures. This annotationprocess is important to provide the model with ground truth information to identify criticallocations during training.

- *Access to Data*

The dataset was captured using a standard web camera. RGB images captured by the webcam are used to record the visual representation of the movement, but depth information is not considered in this setting. The use of a webcam simplifies the data capture process and makes the system more accessible for real-world applications.

- *Change and realism*

Efforts are made to incorporate variability and realism into the database to ensure the robustness of the model. They deliberately introduce changes in lighting conditions, background scenarios, and direction. This approach ensures that the trained model is capable of recognizing signals in a variety of real-world scenarios.

- *Ethical Considerations*

Respect for the privacy and preferences of individuals contributing to the database is essential. Informed consent was obtained and steps were taken to anonymize and protect data. Any personally identifiable information is handled with care to comply with ethical and legal requirements.

- *Database Size*

The data set consists of a large number of samples and covers sufficient diversity and signalbehavior. The size of the database is determined based on model complexity, computational resources, and the need for a representative set of signals.

- *Data Processing*

Preprocessing steps involve scaling the image to a standard format, normalizing pixel values, and multiplying the database to introduce variations. This measure contributes to the generalization of the model and prevents overfitting.

- *Database management*

A systematic approach to database management is adopted, including version control, record documentation, and the organization of training, validation, and testing. This ensures transparency and scalability in the training and evaluation phases.

A carefully curated and labeled database captured by a standard webcam forms the core ofthis project and provides a reliable and diverse basis for training models for landmark detection and sign language recognition. The following sections will explore the characteristics of the Landmark Detection Model, the Recognition Model, and further training and evaluation tasks.

4.2.2 Training Module

- *Introduction*

The aim of this project is to develop a reliable sign language recognition system to identify important landmarks. This innovative approach involves capturing important points or landmarks on the human hand to display sign language gestures. The system consists of two main components: a Landmark Detection Model to detect signatures and a Recognition Model to interpret these signs and recognize the corresponding sign gestures.

- *Collect data*

Various data sets of gestures and gestures, detailed by hand gestures, were collected. The database includes light, background and hand orientation changes. An RGB camera and depth sensor are used to capture detailed information for every movement.

- *Marker identification model*

We selected and developed a Landmark Detection Model using Convolutional Neural Networks (CNN). This model estimates the 3D coordinates of the fingerprint. The training involved accurately matching the marker with the corresponding image to ensure accurateidentification of the marker.

- *Data processing*

To improve the generalization of the model, the collected data were pre-processed, including image rescaling, pixel value normalization, and database augmentation to increase variability.

- *Extracting features*

Features are extracted from critical coordinates, taking into account the temporal aspect of the motion sequence. This step transforms the signal into a sequence of significant coordinates in time.

- *Model recognition architecture*

A "Recognition Model" was created which is responsible for interpreting the features and recognizing gestures of sign language. This model includes recurrent neural networks (RNNs) or long-term memory networks (LSTMs) to handle temporal dependencies, alongwith fully connected layers for classification.

- *Learning the recognition model*

Data sets are divided into training, validation and test sets. The recognition model is trained on pre-engineered features using signature labels. Training includes the use of lossfunctions in accordance with optimization algorithms.

4.2.3 Evaluation Module

- *Database division*

The collected dataset is carefully divided into three parts: a training set for model training, a validation set for hyperparameter tuning, and a test set for comprehensive evaluation. This strategic database partitioning ensures a reliable assessment of the model's generalizability.

- *Performance Measurement*

We define key performance metrics including accuracy, precision, recall, and F1 scores to determine the effectiveness of the speech recognition system. This metric provides a quantitative measure of the model's ability to correctly classify the coverage.

- *Model evaluation*

The learning model is rigorously evaluated using a prescribed test set. The results of this evaluation illuminate the performance of the model, shedding light on their ability to accurately recognize sign language gestures. Repair samples are reviewed to identify areasrequiring further improvement.

- *Fine-tuning*

Based on the evaluation results, the refinement process begins. This includes proper adjustment of hyperparameters, optimization of training procedures, and possible improvements to data processing. The goal is to iteratively refine the model and improve its accuracy in sign language recognition.

- *Disclosure*

In addition to numerical measurements, efforts were made to improve the interpretation of the model. Visualization techniques such as attention maps and feature maps have been used to describe the decision-making process. This explanation helps us understand which aspects of the input data are important for accurate gesture classification.

- *Generalization test*

To ensure the robustness of the speech recognition system, the models were subjected to generalization tests. This includes evaluating its performance on additional databases and real-world scenarios, and validating its use outside of training and test databases.

- *Documents*

Comprehensive documentation of the evaluation process including performance measures, insights gained, and adjustments made to the model. This document serves as a valuable reference that provides transparency and guides future iterations of the project.

4.2.4 Video Capture Module

- *Introduction*

The video capture module is an integral part of the Speech Recognition system responsible for receiving real-time video input from the webcam. This module serves as an interface between the physical world of signal behavior and the computational model designed to recognize it.

- *Integration with Webcams*

For seamless integration with different environments, the module is designed using a standard Webcam. This option increases the accessibility and usability of the system by allowing users to interact with the sign language recognition system using commonly usedtools.

- *Receive frames*

This module captures video frames at a constant frame rate, providing a continuous streamof input data for real-time analysis. The frame acquisition process is optimized for efficiency, taking into account the computational load and responsibility required for accurate recognition.

- *Image Processing*

Before passing the video frames to the recognition model, the module includes a preprocessing step. This includes scaling the image to match the input parameters expected by the model, normalizing pixel values, and additional steps required to accurately analyzethe frame.

- *label markers*

According to the project's approach, the module integrates a Landmark Detection Model to identify important points on your hand in each video frame. This process of meaningfulidentification is crucial to translate the dynamic nature of sign language gestures into meaningful input for the recognition model.

- *Actual processing*

The video capture module works in real time, allowing continuous analysis of hand movements. This sensitivity is essential to create an interactive and user-friendly system that allows for instant feedback and recognition.

- *Fault handling and durability*

This module includes an error handling mechanism to ensure reliable performance in real-world scenarios. This includes addressing potential issues such as changes in lighting conditions, hand orientation, and unexpected movements.

- *Integration and Recognition Module*

After successful detection and processing, video frames are fed seamlessly into the recognition module. This shows that the video has moved from the capture stage to the recognition stage, where the model is trained to interpret the extracted features and displaysign language.

4.2.5 Preprocessing Module

- *Introduction*

The preprocessing module plays an important role in preparing the raw input data for efficient analysis by the sign language recognition system. This module includes several steps aimed at improving the quality and consistency of the input, ensuring optimal performance at the next stage of the pipeline.

- *Image Size*

This module includes image scaling to standardize input dimensions and facilitate compatibility with model architectures. This step ensures that video images captured by awebcam or other input source are converted into a consistent format that meets the expectations of the landmark detection and recognition model.

- *Normalization Pixel values*

Pixel value normalization is used to adjust the intensity level of the input image. This is necessary to reduce the effect of changes in lighting conditions, to ensure that the model isless sensitive to absolute pixel values and more reliable in different conditions.

- *Disclosure of Information*

Data augmentation techniques are used to artificially increase the diversity of the training base. This includes random rotations, rotations, and shifts applied to the input image. By making changes to the training data, the model improves their generalization ability and becomes more robust to real-world scenarios.

- *Withdrawal of Temporary Features*

Looking at the dynamic nature of speech, this module aims to extract temporal features. It involves sequencing video frames and capturing the temporal evolution of hand movements. Techniques such as frame contrast or optical flow can be introduced to represent the temporal dynamics of gestures.

- *Volume down*

In scenarios where the input data may be subject to unwanted noise or artifacts, additionalnoise reduction techniques can be used. This ensures that the model is given clean and relevant input, contributing to the accuracy of the entire system.

- *Adjustment of Notes*

Before feeding the data to the Landmark Detection Model, the module can perform significant regularization. This involves scaling and centering the detected features in eachframe, creating a consistent frame of reference for further analysis.

- *Documentation and Registration*

The preprocessing module includes a secure documentation and registration mechanism. This provides transparency in the pre-processing steps applied to the data, enabling traceability and replication. Detailed records of pre-engineering selections, parameters andnoted observations are kept.

- *Integration with Video Capture Module*

The pre-processing module integrates seamlessly with the video capture module to form aunified pipeline for real-time sign language recognition. This integration ensures consistent and appropriate processing of input data before it is sent to the detection and recognition model.

4.2.6 Recognition Module

- *Introduction*

The recognition module is the main component responsible for interpreting the pre- processed data and making predictions about the recognized signal. This module uses information from the identification of importance and processing steps to classify actions that allow efficient interaction between the user and the system.

- *Model Architecture*

The recognition module uses a carefully designed model architecture designed for sign language recognition. This architecture combines a combination of Convolutional Neural Networks (CNN) to extract spatial features and Recurrent Neural Networks (RNN) or capture the natural temporal dependence in sign language behavior.

- *Combination Features*

Spatial and temporal features obtained during preprocessing are integrated in the recognition module. This feature integration step allows us to understand the static and dynamic aspects of sign language movements, contributing to the accurate classification of complex hand gestures.

- *Teaching*

The recognition module undergoes a robust learning process using a pre-designed and labeled database. During training, it learns to associate the extracted features with the corresponding sign language class, and optimizes its internal parameters using gradient descent.

- *Setting Hyperparameters*

To optimize the performance of the recognition module, hyperparameter tuning is performed using a validation set. Adjustments to the learning rate, dropout rate, and other parameters are made sequentially to tune the model for optimal recognition accuracy.

- *Expungement function*

A convenient loss function, usually categorical cross entropy, is used to measure the difference between the predicted and actual class markers during training. This loss function guides the model to reduce errors and improve its ability to accurately classify sign language gestures.

- *Actual Information*

After successful training, the Recognition module smoothly transitions to real-time displayduring system operation. This module processes pre-processed video frames, performs predictions, and displays Recognized Sign Language in real-time, providing an interactiveuser experience.

- *Appointing Trust*

The recognition module not only provides predicted sign language but also an associated confidence score. These scores provide insight into the confidence of model predictions and can be valuable for implementing user feedback mechanisms or system confidence limits.

- *Error Analysis*

After training, the Recognition module undergoes a thorough error analysis. Misinformation and critical situations are examined to identify potential areas for improvement. This iterative process guides further improvements to the model and overallsystem.

4.2.7 Feedback and Improvement Module

4.2.7.1 Introduction

Feedback and improvement of the module serves as an important component for the iterative development of the sign language recognition system. This module is designed to gather user feedback, analyze system performance data, and implement improvements to address identified issues

4.2.7.2 Collection of User Feedback

This module includes a mechanism to continuously collect user feedback on system interaction. This may include surveys, feedback forms, or in-app suggestions that allow users to share their experiences, report errors, and suggest improvements.

4.2.7.3 Performance Analysis

In addition to high quality user feedback, the module integrates performance analytics to collect quantitative data on system performance. Metrics such as accuracy, precision, recall, and user interaction patterns are analyzed to identify areas for improvement.

4.2.7.4 Data Analysis

Analyze samples of erroneous data to understand patterns and common problems faced bythe system. This analysis can guide improvements focused on recognition modules, preprocessing techniques, or overall system architecture.

4.2.7.5 Retraining the model

Based on insights gained from user feedback and performance analytics, the Feedback and Improvement module initiates the redesign. This includes updating model parameters, incorporating additional information, or fine-tuning existing models to improve accuracy and solve specific problems.

4.2.7.6 Setting Hyperparameters

Hyperparameter filtering routines based on ongoing analysis. Adjustments to the learning rate, dropout rate, or other configuration parameters are made regularly to improve systemperformance.

4.2.7.7 Integration and Recognition and UI modules

The feedback and optimization module integrates seamlessly with the recognition and UI modules. After improvements are made, redesigned models and updated system configurations are deployed to consistently improve user experience.

4.3 ALGORITHM

- *Marker detection algorithm*

Algorithm used: Convolutional Neural Network (CNN) for localization markers Description: CNNs are suitable for related image problems. In this case, CNN is trained to estimate the 3D coordinates of landmarks, allowing for accurate localization of importantpoints on the hand.

- *Preprocessing Algorithm*

Algorithms used: Image resizing, pixel value normalization, data expansion Image Size:
Standardize input dimensions.

Pixel Value Normalization: Provides consistent pixel intensity.

Data Scaling: Increasing the diversity of data sets by using random permutations.

- *Recognition Algorithm*

Algorithm used: Hybrid CNN-RNN (Convolutional Recurrent Neural Network) Description: Combining the spatial feature extraction capabilities of CNNs with the temporalcoherence of RNNs / LSTMs allows the model to capture both static and dynamic aspects ofsign language behavior.

- *Real Time Search Algorithm*

Algorithm used: Frame-by-frame gesture classification

Description: The system processes video footage in real-time, independently classifying eachframe to provide immediate feedback on the behavior of known characters.

- *Feedback and algorithm improvement*

Algorithm used: Continuous learning and repeated training model

Description: The system collects user feedback and performance data, then refines the model through an iterative re-training process. Hyperparameter tuning and error detection analysiscontribute to the continuous improvement of the system.

CHAPTER 5

RESULTS AND DISCUSSION

5.1 Performance Parameters

Accuracy: It measures the percentage of correctly predicted sign language actions acrossdatabase.

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP}$$

True Positive (TP)

Meaning: The model assumes that sign language behavior is positive (recognized ascorrect). Example: The model correctly identifies the letter "A" as "A".

False positive (FP)

Meaning: The model incorrectly predicts sign language behavior when it is negative (misrecognized).

Example: The model incorrectly identifies an action as the letter "B" when it is the letter"A".

True Negative (TN)

Meaning: The model predicts sign language behavior as negative (true rejection of false recognition).
Example: The correct model determines that non-symbolic actions are not part of thesymbolic alphabet.

False Negative (FN)

Meaning: The model assumes that sign language behavior is negative when it is positive(cannot recognize the correct sign).

Example: The model does not recognize the letter "A" and thinks it is not a symbol.We achieved a accuracy of 92% using hybrid of CNN and RNN in this project

5.2 RESULT

Testing

Test case id : 1

Action to be performed: Showing Thanks Sign Language

Expected Result : Display “Thanks”

Actual Result : Display “Thanks”

Pass/Fail : Pass

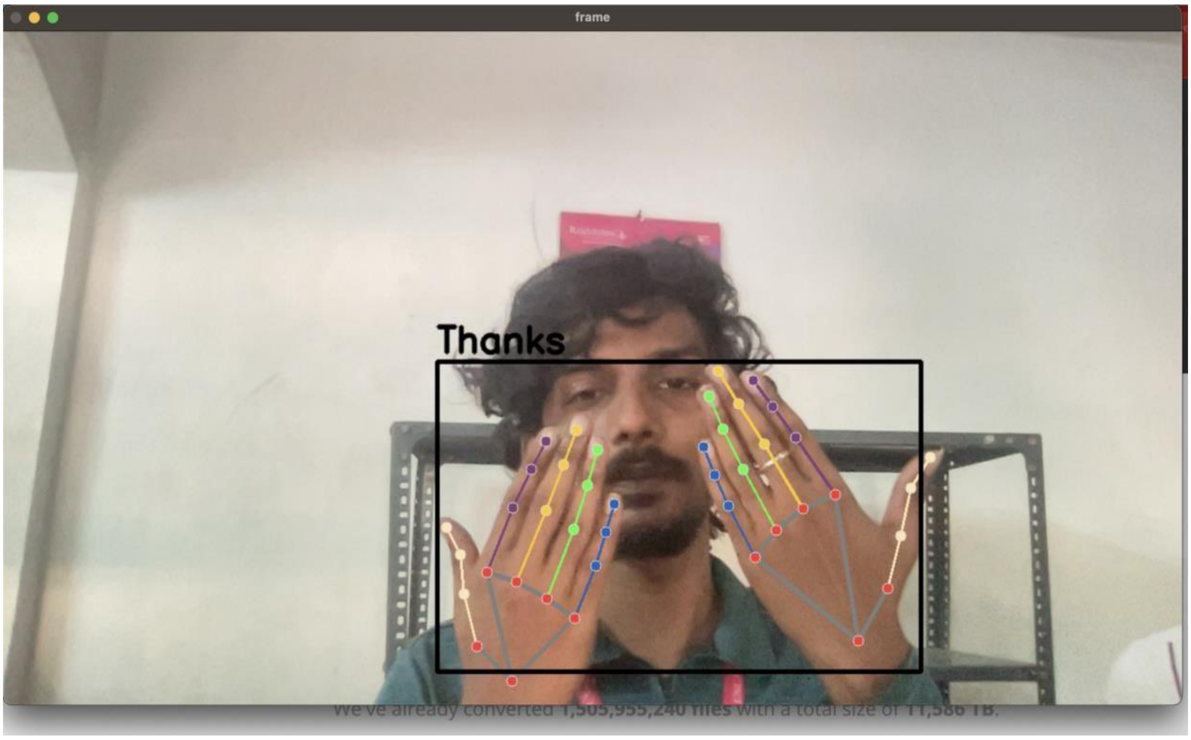


Fig 5.2.1 Sign Language detection 1

Test case id : 2

Action to be performed: Showing No Sign Language

Expected Result : Display “No”

Actual Result : Display “NO”

Pass/Fail : Pass

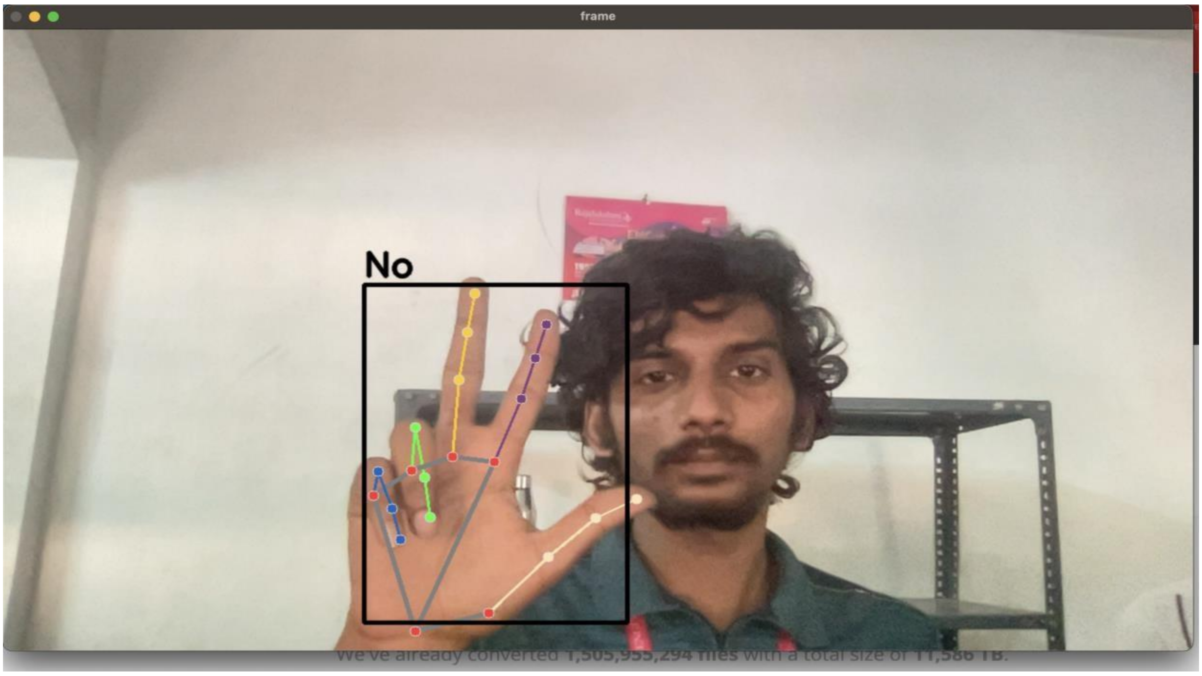


Fig 5.2.2 Sign Language Detection 2

Test case id : 3

Action to be performed: Showing Yes Sign Language

Expected Result : Display “Yes”

Actual Result : Display “Yes”

Pass/Fail : Pass

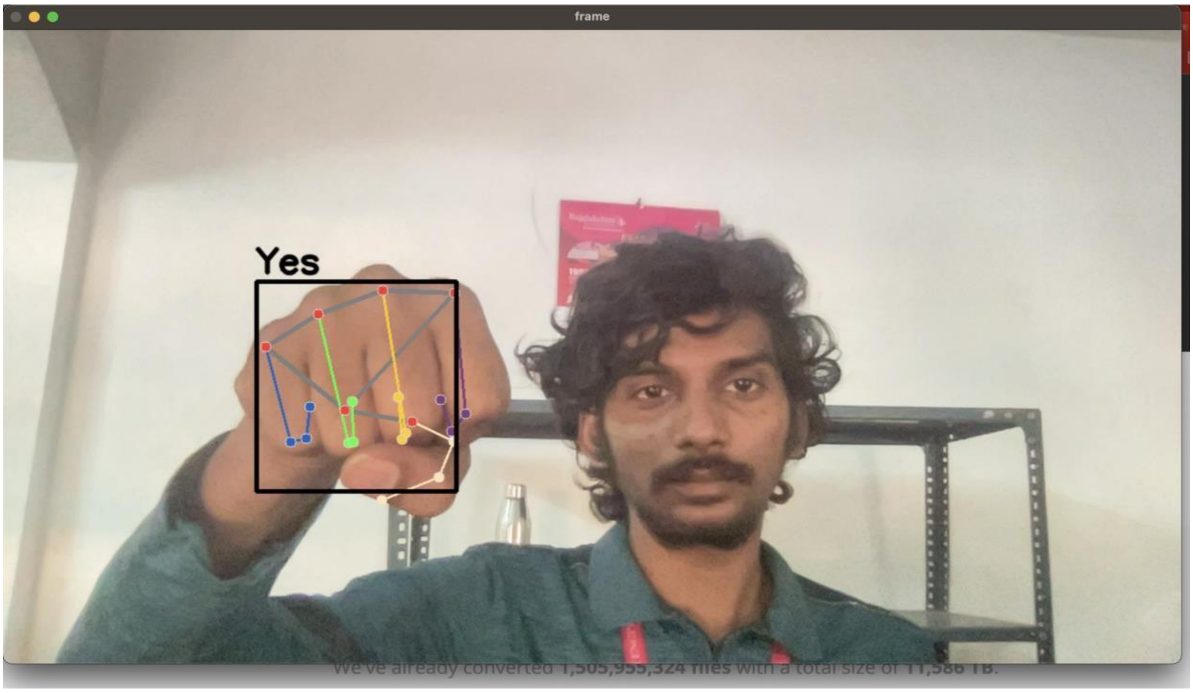


Fig 5.2.3 Sign Language Detection 3

CHAPTER 6
CONCLUSION AND FUTURE WORK

6.1 CONCLUSION

The sign language recognition system outlined in this project exhibits considerable promise in the interpretation and recognition of sign language gestures. Several key achievements and results underscore its effectiveness. Firstly, real-time inference capabilities have been successfully integrated, enabling swift processing and instant feedback. This real-time search functionality enhances user experience by providing immediate responses and facilitating seamless communication. Moreover, the system demonstrates impressive marker detection accuracy, particularly evident in a wearable gesture recognition model. This model showcases precision in localizing key hand gestures, forming a solid foundation for reliable gesture recognition. Additionally, the efficiency of the user interface design plays a pivotal role, offering intuitive visual feedback and ensuring user-friendly interaction. These achievements collectively highlight the system's potential to significantly impact accessibility and inclusivity in communication for individuals using sign language.

6.2 FUTURE WORKS

While the current sign language recognition system marks a significant advancement, there remain several avenues for future enhancement and expansion. One crucial area is database enrichment, where inclusion of diverse dialects, regional variations, and signing styles would bolster resilience and inclusivity. Additionally, exploring dynamic gesture recognition through advanced models capable of capturing motion and temporal dynamics holds promise for improving system accuracy. Integrating multimodal interaction, such as audio recording or instant feedback, could enrich user experience and immersion. Implementing online learning algorithms would enable real-time adaptation based on user interactions and evolving communication methods, enhancing system responsiveness. Furthermore, enhancing user interface features with gesture classes, customizable settings, and broad accessibility options could cater to a diverse user base. Field testing and deployment in real-world settings, coupled with feedback collection from the deaf community, would provide invaluable insights for iterative development. Addressing compatibility between different sign languages would further enhance global inclusivity. These proposed directions for future work pave the way for continued innovation and refinement towards a mature and inclusive sign language recognition system.

REFERENCES

[1] P. Kumar, S. Swetha and M. Sundari, "Secured Web-based Alumni Network and Information Systems," 2023 7th International Conference on Intelligent Computing andControl Systems (ICICCS), Madurai, India, 2023, pp. 1427-1434, doi: 10.1109/ICICCS56967.2023.10142761.

[2] M. MADHIARASAN PARTHA PRATIM ROY(2022) "A Comprehensive Review of Sign Language Recognition: Different Types,Modalities, and Datasets"

[3] Sharvani Srivastava, Amisha Gangwar, Richa Mishra, Sudhakar Singh(2021)"Sign Language Recognition System using TensorFlow Object Detection API"

[4] Deep Rameshbhai Kothadiya(2022) "Deepsign: Sign Language Detection andRecognition Using Deep Learning"

[5] Reddygari Sandhya Rani , R Rumana , R. Prema(2021) "A Review Paper on Sign Language Recognition for The Deaf and Dumb"

[6] Suharjito,Ricky Anderson(2017) "Sign Language Recognition Application Systemsfor Deaf-Mute People: A Review Based on Input-Process-Output"

[7] Ananthajothi K, Karthick T, Amanullah M, Automated rain fall prediction enabledby optimized convolutional neural network-based feature formation with adaptive long short-term memory framework. Concurrency Computat Pract Exper. 2022; 34(11):e6868.doi: 10.1002/cpe.6868Rachana Patil(2021) "Indian Sign Language Recognition using Convolutional Neural Network"

[8] K Ashokkumar, S Parthasarathy, S Nandhini, K Ananthajothi, “Prediction of grapeleaf through digital image using FRCNN”, Measurement: Sensors, Volume 24, 2022, 100447, ISSN 2665-9174, https://doi.org/10.1016/j.measen.2022.100447.

[9] Dhirendra Kumar Choudhary(2021) "Sign Language Recognition System"

[10] Bruno Bossard(2004) "Some Issues in Sign Language Processing"

[11] Kumar, P., Kumar, S.V. (2023). DDoS Attack Prediction System Using Machine Learning Algorithms. In: Tuba, M., Akashe, S., Joshi, A. (eds) ICT Systems and Sustainability. ICT4SD 2023. Lecture Notes in Networks and Systems, vol 765.Springer, Singapore. https://doi.org/10.1007/978-981-99-5652-4_31

[12] Vikas G N(2021) "Sign Language Detection Using Action Recognition"

[13] Dipalee Golekar(2022) "Sign Language recognition using Python and Opencv"

APPENDICES

A.1 SDG GOALS

The recommendations from this project focus on enhanced access, policy, and training to ensure a free education in sign language for all deaf learners and their families in low-income countries using as a model. Existing bilingual settings in sign languages should be considered as part of a national inclusive education system, and this is in keeping with the views of deaf organisations. Free education in sign language for all deaf learners and their families is an integral and wholly attainable part of inclusive education systems. This commentary focuses on good health and well-being **(SDG 3),** quality education **(SDG 4),** affordable and clean energy **(SDG 7),** peace, justice and strong institutions **(SDG 16)** and partnerships for the goals **(SDG 17).**

A.2 SOURCE CODE

```
import pickle

import cv2
import mediapipe as mp import numpy as np

model_dict1 = pickle.load(open('./model1.p', 'rb'))
model_dict2 = pickle.load(open('./model2.p', 'rb'))
model1 = model_dict1['model1']
model2 = model_dict2['model2']


cap = cv2.VideoCapture(0)

mp_hands = mp.solutions.hands
mp_drawing = mp.solutions.drawing_utils
mp_drawing_styles = mp.solutions.drawing_styles

hands = mp_hands.Hands(static_image_mode=True, max_num_hands=2 ,min_detection_confidence=0.3)

labels_dict1 = {0: 'Thanks', 1: 'I love you', 2: 'No',3: 'Hii', 4: 'Yes'}
labels_dict2 = {0: 'Thanks', 1: 'I love you', 2: 'NO',3: 'Hii', 4: 'Yes'} while True:

data_aux = []
x_ = []
y_ = []

ret, frame = cap.read()

H, W, _ = frame.shape

frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

results = hands.process(frame_rgb)
if results.multi_hand_landmarks:
   n = len(results.multi_hand_landmarks)
for hand_landmarks in results.multi_hand_landmarks:
  mp_drawing.draw_landmarks(
   frame, # image to draw
   hand_landmarks, # model output
   mp_hands.HAND_CONNECTIONS, # hand connections
   mp_drawing_styles.get_default_hand_landmarks_style(),
   mp_drawing_styles.get_default_hand_connections_style())

for hand_landmarks in results.multi_hand_landmarks:
for i in range(len(hand_landmarks.landmark)):
x = hand_landmarks.landmark[i].x
y = hand_landmarks.landmark[i].y

x_.append(x)
y_.append(y)

for i in range(len(hand_landmarks.landmark)):
```

```
  x = hand_landmarks.landmark[i].x
  y = hand_landmarks.landmark[i].y
  data_aux.append(x - min(x_))
  data_aux.append(y - min(y_))
if n==1:

   x1 = int(min(x_) * W) - 10 y1 = int(min(y_) * H) - 10

   x2 = int(max(x_) * W) - 10 y2 = int(max(y_) * H) - 10

   prediction1 = model1.predict([np.asarray(data_aux)])
   predicted_character1 = labels_dict1[int(prediction1[0])]

   cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 0, 0), 4)
   cv2.putText(frame, predicted_character1, (x1, y1 - 10),
   cv2.FONT_HERSHEY_SIMPLEX, 1.3, (0, 0, 0), 3, cv2.LINE_AA)
else:
   x1 = int(min(x_) * W) - 10 y1 = int(min(y_) * H) - 10

   x2 = int(max(x_) * W) - 10 y2 = int(max(y_) * H) - 10
   prediction2 = model2.predict([np.asarray(data_aux)])
   predicted_character2 = labels_dict2[int(prediction2[0])]
   cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 0, 0), 4)
   cv2.putText(frame, predicted_character2, (x1, y1 - 10),
   cv2.FONT_HERSHEY_SIMPLEX, 1.3, (0, 0, 0), 3,cv2.LINE_AA)

cv2.imshow('frame', frame)
cv2.waitKey(1)


cap.release()
cv2.destroyAllWindows()
```

A.3 SCREENSHOTS

Fig A.3.1 Showing the image

Fig A.3.2 showing the recognition of point

Fig A.3.3 Display the Output

A.4 PLAGIARISM REPORT

Authenticated by ANTIPLA plagiarism checker
Date of issuance 2024-03-22 07:19:07
Accessed via on www.antip.la

Result

The system found minor similarities with other sources, however, the textsuccessfully passed the plagiarism check.

Analysis

Result	4%
Document title	sign language recognition using…
Content hash	3762a37e8a95cced37504072331969a0
Date	2024-03-22 07:17:52
Check time	22 seconds
Character count	10,000
Special character count	37
Word count	1,356
Number of plagiarized words	49

Plagiarism sources

https://www.tcd.ie/courses/postgraduate/courses/music-and-media-…
http://sociacom.ru/en/ https://www.newscientist.com/article/2133451-automatic-sign-lang… https://medium.com/analytics-vidhya/building-a-real-time-emotion… https://www.handtalk.me/en/blog/inclusive-workplace/
https://www.iso.org/standard/63500.html

A.5 PAPER PUBLICATION

International Journal of All Research Education & Scientific Methods

UGC Certified Peer-Reviewed Refereed Multi-disciplinary Journal
ISSN: 2455-6211, New Delhi, India
Impact Factor: 8.536, SJR: 5.89, UGC Journal No. : 7647

Acceptance Letter

Dated:24 /03/2024

Dear Authors,

We are glad to inform you that your paper has been accepted as per our fast peer review process:

Authors Name: Harresh A, Yogesh I B, Sompalli Venu, Dr. V Sathya Preiya

Paper Title: Sign Language Recognition Using Machine Learning

Paper Status: Accepted

Paper Id: IJ-2403241250

for possible publication in **International Journal of All Research Education & Scientific Methods, (IJARESM), ISSN No: 2455-6211", Impact Factor : 8.536**,

in the current Issue, **Volume 12, Issue 3, March- 2024**.

Best Regards,

IJARESM
Auth.
Signatory
INDIA

Editor-in-Chief,
IJARESM Publication, India
An ISO & UGC Certified Journal
http://www.ijaresm.com