RSET
RAJAGIRI SCHOOL OF
ENGINEERING & TECHNOLOGY
(AUTONOMOUS)

Mini Project Report On

Sign Language Translation System

Submitted in partial fulfillment of the requirements for the award of the degree of

Bachelor of Technology

in

Computer Science & Engineering

By

Aarathi Nair (U2003002)
Anitta Mariya Shaji (U2103034)
Anjala Binu (U2103035)
Anushri Dilip (U2103043)

Under the guidance of

Ms. Anita John

Department of Computer Science & Engineering
Rajagiri School of Engineering & Technology (Autonomous)
(Affiliated to APJ Abdul Kalam Technological University)
Rajagiri Valley, Kakkanad, Kochi, 682039
May 2024

CERTIFICATE

This is to certify that the mini project report entitled ***"Sign Language Translation System"*** *is a bonafide record of the work done by* ***Aarathi Nair (U2003002), Anitta Mariya Shaji (U2103034), Anjala Binu (U2103035), Anushri Dilip (U2103043)*** *submitted to the APJ Abdul Kalam Technological University in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (B. Tech.) in Computer Science and Engineering during the academic year 2023-2024.*

Ms. Anita John
Assistant Professor
Dept. of CSE
RSET

Mr. Harikrishnan M, Assistant Professor
Ms. Sherine Sebastian, Assistant Professor
Dept. of CSE
RSET

Dr.Preetha K G
Head of Department
Dept. of CSE
RSET

ACKNOWLEDGEMENTS

I wish to express my sincere gratitude towards Dr P. S. Sreejith, Principal of RSET, and Dr. Preetha K.G., Head of the Department of Computer Science and Engineering for providing me with the opportunity to undertake our mini project, "Sign Language Translation System".

I am highly indebted to my project coordinators, **Mr. Harikrishnan M and Ms. Sherine Sebastian**, Assistant Professor, Department of Computer Science and Engineering for their valuable support.

It is indeed my pleasure and a moment of satisfaction for me to express my sincere gratitude to my project guide **Ms. Anita John** for her patience and all the priceless advice and wisdom she has shared with us.

Last but not the least, I would like to express my sincere gratitude towards all other teachers and friends for their continuous support and constructive ideas.

Aarathi Nair

Anitta Mariya Shaji

Anjala Binu

Anushri Dilip

SIGN LANGUAGE TRANSLATION SYSTEM

Our project aims to create a computer system that can easily translate between American Sign Language (ASL) and Indian Sign Language (ISL) using advanced technology called Natural Language Processing (NLP). Even though ASL and ISL are similar, they have different words, sentence structures, and rules, making translation tricky. We will collect images of people using ASL and ISL along with written descriptions of what they are saying as our dataset. Then, we will organize and clean up this data to make it useful for training our computer system. We will make a special kind of computer program that can understand the unique patterns and rules of ASL and ISL. We will train our program using the data we collected, teaching it how to translate accurately between ASL and ISL. Once our program is trained and tested, we will make it easy for people to use, whether on a website or a phone app. This way, anyone can quickly translate between ASL and ISL whenever they need to. By creating this system, we hope to make communication easier for people who use sign language and contribute to improving technology for sign language translation everywhere.

Contents

List of Figures

List of Abbreviations

SLTS - Sign Language Translation System

ASL - American Sign Language

ISL - Indian Sign Language

Chapter 1

Introduction

1.1 Background

The development of a Sign Language Translation System (SLTS) represents a critical advancement in bridging communication barriers between individuals who are deaf or hard of hearing and those who communicate primarily through spoken language. In today's world, the need for effective communication across diverse linguistic and sensory modalities is more pressing than ever.

Communication barriers faced by the deaf and hearing-aided individuals are prevalent across various domains of daily life. In educational settings, students with hearing impairments often encounter challenges in fully participating in classroom activities and engaging with their peers. Likewise, in professional environments, deaf employees may struggle to access crucial information conveyed through spoken communication, hindering their productivity and integration within the workforce. Moreover, social interactions and accessing public services can be daunting tasks for individuals who rely on sign language as their primary mode of communication.

The development of an SLTS holds immense significance in addressing these challenges and empowering individuals with hearing impairments to communicate effectively in various contexts. By using machine learning, an SLTS can provide real-time translation between sign languages. This innovation has the potential to revolutionize communication accessibility for deaf individuals, enabling them to participate more fully in educational, professional, and social settings.

Furthermore, an SLTS promotes inclusivity and diversity in society. By facilitating seamless communication between individuals who use different sign languages, it fosters greater understanding and empathy across linguistic and cultural boundaries.

In conclusion, the development of a Sign Language Translation System represents a

transformative advancement in promoting communication accessibility and inclusivity for individuals who are deaf or hearing impaired thereby enhancing the quality of their life.

1.2 Problem Definition

The problem addressed by the Sign Language Translation System is to develop a software that accurately detects and translates American Sign Language (ASL) gestures into Indian Sign Language (ISL) and vice versa, ensuring seamless communication between users of different sign languages.

1.3 Scope and Motivation

The project's scope is dedicated to creating a sign language translation system specifically designed to translate between American Sign Language (ASL) and Indian Sign Language (ISL) using hand gestures. By using machine learning technologies, the system will accurately interpret gestures in ASL and convert them into equivalent gestures in ISL, and vice versa. This initiative aims to facilitate seamless communication between ASL and ISL users, breaking down language barriers within the deaf and hard of hearing communities of the United States and India. The system will prioritize ease of use through an intuitive user interface, ensuring accessibility for all users. Continuous updates and improvements will be made based on user feedback, with a goal to enhance understanding and inclusivity between these two sign language communities.

The motivation behind developing a sign language translation system between American Sign Language (ASL) and Indian Sign Language (ISL) stems from a profound need to bridge the communication divide between two of the world's most vibrant deaf communities. With thousands of miles and cultural differences separating these communities, the system aims to foster a deeper understanding and connection by eliminating linguistic barriers. By leveraging advanced technology to translate sign languages in real-time, we envision empowering individuals with the ability to communicate freely and effectively, regardless of their native sign language. This initiative not only promotes inclusivity and accessibility but also celebrates the rich diversity of sign languages and their cultures. Ultimately, the project aspires to be a stepping stone towards a more interconnected and supportive global deaf community, where no individual is left isolated due to language

limitations.

1.4 Objectives

1. Develop Real-time Translation: Create a robust system capable of translating between American Sign Language (ASL) and Indian Sign Language (ISL) in real-time, ensuring minimal delay to facilitate seamless communication.

2. Achieve High Translation Accuracy: Aim for a high level of accuracy in gesture recognition and translation between ASL and ISL, reducing misunderstandings and promoting clear communication.

3. Enhance Accessibility: Design an intuitive and user-friendly interface that is easily navigable by all users, including those with no prior experience with translation technologies, to ensure the system is accessible to a broad audience.

4. Promote Cultural Exchange: Foster cultural exchange and understanding between the ASL and ISL communities by providing a tool that breaks down language barriers, allowing for the sharing of knowledge, culture, and experiences.

5. Facilitate Educational Opportunities: Enable greater educational and learning opportunities for users by providing a platform that can be used for self-learning, classroom learning, and cross-cultural educational initiatives.

1.5 Challenges

Developing a Sign Language Translation System that translates between Indian and American Sign Languages presents significant challenges due to the linguistic and cultural differences between the two sign languages. These challenges include variations in handshapes and gestures as well as differences in cultural references and idiomatic expressions, requiring sophisticated algorithms to accurately capture and translate the nuances of each language.

1.6 Assumptions

1. Sign Language Recognition Accuracy:The accuracy and effectiveness of the sign language translation software heavily rely on the availability of high-quality training data for both Indian Sign Language (ISL) and American Sign Language (ASL). Sufficient and diverse datasets for both languages are available for model training. This assumption could be incorrect if the datasets are limited or of poor quality, affecting the accuracy of translation.

2. Hardware Requirements: The performance of the sign language translation software may be affected by the hardware it runs on, particularly the processing power and memory. The hardware requirements, such as CPU specifications and memory capacity are sufficient to support real-time translation without significant delays. If the hardware is insufficient, it could lead to performance issues or slower translation speeds.

3. Integration with External Components: The software relies on external components such as cameras for gesture recognition and displays for presenting translated sign language animations.Assume that seamless integration with these external components is possible, and they provide reliable input and output for the translation process. Compatibility issues or hardware limitations with external components could impact the functionality of the software.

4. Cultural and Linguistic Sensitivity: Sign language translation involves nuances and cultural variations that must be accurately captured.Incorrect assumptions about these factors could lead to misunderstandings or misinterpretations in translation.

1.7 Societal / Industrial Relevance

The development of a sign language translation system between American Sign Language (ASL) and Indian Sign Language (ISL) represents a significant stride towards inclusivity and accessibility in our increasingly interconnected world. This project addresses the urgent need for effective communication tools within and between deaf communities globally. Its societal and industrial relevance extends beyond the immediate benefit of

facilitating communication, touching on areas as diverse as education, employment, cultural exchange, and technological advancement. By exploring the multifaceted impact of this project, we gain insight into its potential to not only change individual lives but also to influence society and industry at large.

The first and perhaps most direct impact of the ASL-ISL translation system is its ability to bridge communication gaps. For deaf individuals who use either ASL or ISL, the barrier to communicating with peers, educators, and employers who use a different sign language can be substantial. This system promises to tear down these barriers, fostering a level of social inclusion and integration previously challenging to achieve. By enabling seamless conversation between ASL and ISL users, it paves the way for a society where the deaf and hard of hearing can engage more fully in social, cultural, and professional environments.

On the educational front, the translation system holds the promise of dramatically enhancing access to learning. Deaf students and educators can share knowledge and resources across linguistic divides, enriching the educational experience and fostering a more inclusive academic community. This tool could also facilitate the development of cross-cultural educational programs, allowing students to learn about and understand diverse cultures and languages, thereby broadening their educational horizons and preparing them for a globalized world.

In the realm of employment, the project's significance cannot be overstated. Improved communication capabilities can open new doors for deaf individuals in the workforce, allowing them to pursue careers in fields where language barriers might have previously excluded them. For industries, this translates to access to a broader pool of talent and perspectives, enhancing diversity, innovation, and productivity. Furthermore, businesses that adopt and support such technologies signal their commitment to inclusivity, attracting a wider customer base and improving their social impact.

Cultural exchange is another area profoundly impacted by the ASL-ISL translation system. By facilitating direct communication between deaf individuals from different linguistic backgrounds, the system encourages the sharing of cultural experiences, traditions, and stories. This exchange not only enriches individual lives but also fosters mutual understanding and respect among communities, contributing to a more cohesive and tolerant society.

Finally, the project contributes significantly to the advancement of assistive technologies. By pushing the boundaries of what is possible in real-time sign language translation, it sets a new standard for accessibility tools, encouraging further innovation in the field. This progress is crucial for building a world where technology serves to empower all individuals, regardless of their abilities, promoting independence and enhancing the quality of life for those with disabilities.

In conclusion, the development of a translation system between ASL and ISL is a testament to the power of technology to foster inclusivity, understanding, and accessibility. Its impact spans societal and industrial spheres, offering profound benefits in terms of communication, education, employment, cultural exchange, and technological advancement. As this project moves forward, it holds the promise of not just bridging the gap between two sign languages but also of contributing to a more inclusive and connected world for the deaf and hard of hearing communities and beyond.

1.8 Organization of the Report

Chapter 2 deals with Software Requirements Specification (SRS) document which outlines the functional and non-functional requirements of the SLTS. It introduces the purpose and scope of the document, detailing the system's functional requirements, including user interactions and translation capabilities. Non-functional requirements such as performance, security, and usability considerations are also specified.

The "System Architecture and Design" section provides an overview of the project's objectives, emphasizing real-time translation between American Sign Language (ASL) and Indian Sign Language (ISL). It outlines the architectural design, dataset considerations, methodology, and algorithms used for translation. Additionally, it discusses user interface and database design, implementation strategies, module division, and work schedule, ensuring a comprehensive approach to system development and deployment.

Chapter 2

Software Requirements Specification

2.1 Introduction

2.1.1 Purpose

The software requirements specified in this document pertain to a Sign Language Translation project v1.0. This SRS encompasses the development of software tasked with translating American Sign Language to Indian Sign Language and vice versa. It includes functionalities such as sign language detection, natural language processing, sign language generation, user interface design, and integration with input and output devices.The Sign Language Translation software may constitute only a singular subsystem or component within a broader project or system. Other subsystems or components, such as user management or database integration, might have their own distinct SRS documents elucidating their requirements and functionalities.

2.1.2 Product Scope

Benefits:

This software is beneficial in scenarios where a person from the US who uses ASL wants to convey a message to someone from India who uses ISL. This software converts the message from ASL to ISL and vice versa. It is also advantageous in important events like conferences, meetings, or even concerts where there may only be a person who speaks ASL, and the relevant information is converted from ASL to ISL during these situations.

Corporate Goals:

- **Accessibility:** Ensure that the software contributes to making communication more accessible for individuals who use American Sign Language (ASL) and Indian Sign Language (ISL) by providing accurate and efficient translation services.

- **Inclusivity:** Promote inclusivity by bridging the communication gap between ASL and ISL users, facilitating seamless interaction and understanding among diverse communities.
- **Empowerment:** Empower individuals with hearing impairments by enabling them to communicate effectively and independently in various settings, including educational, professional, and social environments.
- **Innovation:** Continuously innovate and improve the software's capabilities to enhance translation accuracy, speed, and usability, staying at the forefront of technology in sign language translation.

Software Goals:

- **Accurate Translation:** Develop algorithms and models that accurately translate between ASL and ISL, capturing nuances and cultural variations to ensure precise communication.
- **Real-time Translation:** Enable real-time translation capabilities to facilitate instant communication between ASL and ISL users, supporting both synchronous and asynchronous communication modes.
- **User-friendly Interface:** Design an intuitive and user-friendly interface that caters to the needs of both deaf and hearing users, making the software accessible and easy to navigate.

Business Strategies:

1. **Partnerships and Collaborations:** Collaborate with organizations, educational institutions, and government agencies involved in deaf advocacy and sign language education to promote the software and expand its user base.
2. **Targeted Marketing:** Implement targeted marketing strategies to reach out to relevant stakeholders, including deaf communities, sign language interpreters, educators, and healthcare professionals, highlighting the benefits and features of the software.

3. **Community Engagement:** Foster a vibrant user community by hosting online forums, webinars, and events where users can share their experiences, provide feedback, and suggest improvements for the software.

4. **Continuous Improvement:** Commit to ongoing research and development to enhance the software's performance, incorporating user feedback and emerging technologies to stay ahead of competitors and meet evolving user needs.

2.2 Overall Description

2.2.1 Product Perspective

The Sign Language Translation Software is a new, self-contained product aimed at bridging the communication gap between users of Indian Sign Language (ISL) and American Sign Language (ASL). It is not a replacement for existing systems but rather fills a significant need for real-time translation between these two distinct sign languages.It empowers people who use ISL and ASL to communicate seamlessly with each other thereby fostering greater social inclusion and accessibility.Also enables real-time conversations across sign languages.

Origin: The idea for Sign Language Translation Software emerged from the growing recognition of the importance of inclusivity and accessibility in communication technology.While there are existing sign language translation systems, they predominantly focus on translating between spoken language and sign language within the same linguistic context.However, there has been a lack of solutions specifically tailored to translate between different sign languages, such as ISL and ASL.

A Sign Language Translation Software is a self-contained product that operates independently but can also function as a component within larger systems aimed at promoting inclusive communication. This software can be potentially integrated with a larger communication system.

2.2.2 Product Functions

- **Sign Language Recognition:** Recognize and interpret sign language gestures made by the user in real-time using camera input, distinguishing between ASL and ISL gestures.

- **Translation Processing:** Convert recognized gestures from the input sign language (either ASL or ISL) to the corresponding expressions in the target sign language.
- **Gesture Generation:** Generate and display the translated sign language gestures in the target sign language (image and text) to convey the translation to the user.
- **User Interface and Interaction:** Provide a user-friendly interface that allows users to easily start translations, switch between ASL and ISL, and view translated gestures.
- **Accessibility Features:** Accessible for users with disabilities, ensuring wide usability across diverse user groups.

2.2.3 Operating Environment

Hardware Platform:

- **Personal Computers (PCs):** Desktops, laptops, and convertible devices with a range of processing power to handle the translation algorithms efficiently.
- **Mobile Devices:** Smartphones and tablets to enable on-the-go communication for users, leveraging their portability and touch interfaces.

Operating System and Versions:

- **Windows:** Support for Windows 10 and above, ensuring compatibility with a large user base of PC users.
- **macOS:** Compatibility with macOS Catalina and later versions to cater to users who prefer Apple products.
- **Linux:** Compatibility with popular distributions such as Ubuntu, Fedora, and Debian to offer flexibility for users who prefer open-source platforms.

Other Software Components or Applications:

- **Integrated Development Environments (IDEs):** Supported with IDEs such as Visual Studio, Xcode, and IntelliJ IDEA for software developers who may want to incorporate the translation functionality into their applications.

- **Web Browsers:** Compatibility with major web browsers like Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge for web-based applications and communication platforms.

- **Collaboration Platforms:** Compatibility with collaboration platforms such as Zoom, Microsoft Teams, and Google Meet, enabling users to communicate effectively during online meetings and discussions.

2.2.4 Design and Implementation Constraints

- ASL and ISL have distinct grammar and vocabulary. Translating between them requires deep understanding and expertise in both languages. Developers must be sensitive to these differences to ensure accurate translations.

- The system may have hardware limitations such as timing requirements or memory constraints, especially if it is intended for deployment on resource-constrained devices like smartphones or tablets. Developers must optimize the system to run efficiently within these constraints.

- Specific technologies, tools, and databases may be mandated or preferred for development. For instance, the system may need to integrate with existing sign language recognition databases or utilize machine learning algorithms for translation.

2.2.5 Design and Implementation Constraints

Assumptions

- **Sign Language Recognition Accuracy:** The accuracy and effectiveness of the sign language translation software heavily rely on the availability of high-quality training data for both Indian Sign Language (ISL) and American Sign Language (ASL). Sufficient and diverse datasets for both languages are available for model training. This assumption could be incorrect if the datasets are limited or of poor quality, affecting the accuracy of translation.

- **Hardware Requirements:** The performance of the sign language translation software may be affected by the hardware it runs on, particularly the processing power

and memory. The hardware requirements, such as CPU specifications and memory capacity, are sufficient to support real-time translation without significant delays. If the hardware is insufficient, it could lead to performance issues or slower translation speeds.

- **Integration with External Components:** The software relies on external components such as cameras for gesture recognition and displays for presenting translated sign language animations. Assume that seamless integration with these external components is possible, and they provide reliable input and output for the translation process. Compatibility issues or hardware limitations with external components could impact the functionality of the software.

- **Cultural and Linguistic Sensitivity:** Sign language translation involves nuances and cultural variations that must be accurately captured. Incorrect assumptions about these factors could lead to misunderstandings or misinterpretations in translation.

Dependencies

- Dependency on external libraries for machine learning.

- Dependency on software development tools and platforms for coding, testing, and deployment.

- Dependency on operating system compatibility for deployment on various platforms (e.g., Windows, macOS, Linux, mobile devices).

2.3 External Interface Requirements

2.3.1 User Interfaces

The software components for which a user interface (UI) is needed include:

- **Translation Interface:**This is the primary component where users input their ASL or ISL gestures and receive the corresponding translations. The UI should include:

- Input area: A space where users can input ASL or ISL gestures by capturing gestures through a camera.
- Output area: A display area where the translated text or gesture are shown in the target language.
- Settings/Options: Allow users to customize translation preferences language selection.

2.3.2 Hardware Interfaces

- **Camera Interface**
  - Logical Characteristics: The software interacts with the camera to capture video input of sign language gestures.
  - Physical Characteristics: Support for various camera types, including built-in webcams, external cameras, and mobile device cameras.
  - Data Interaction: The software receives video feed from the camera as input data for sign language recognition.
- **Display Interface**
  - Logical Characteristics: The software interacts with the display to present translated sign language gestures and text.
  - Physical Characteristics: Support for various display types including monitors, projectors, and mobile device screens.
  - Data Interaction: The software sends visual output to the display for presentation to the user.
  - Control Interaction: The software may control display settings (e.g., resolution, brightness) to optimize visual output.
- **Input/Output Interfaces**
  - Logical Characteristics: The software interacts with input/output devices such as monitors.
  - Physical Characteristics: Support for various input/output device

- Data Interaction: The software receives input from user interactions and provides output through monitor screens.

Communication Protocols used include:

- The software interfaces with the camera typically uses standard protocols such as USB Video Class (UVC) or DirectShow on Windows platforms.
- The software interfaces with display uses standard graphics APIs provided by the operating system (e.g., DirectX, OpenGL).

2.3.3 Software Interfaces

- **TensorFlow 2.x or PyTorch 1.6+:** Deep learning frameworks for building machine learning models.
- **OpenCV 4.x:** Computer vision library for capturing and processing video streams.
- **Flask 1.x or Django 3.x:** Web frameworks for developing user interfaces and backend services.

These software components provide the necessary tools and libraries for implementing the sign language translation algorithms and user interface.

2.4 System Features

2.4.1 ASL Detection

2.4.1.1 Description and Priority

This feature enables the system to recognize and interpret gestures from American Sign Language (ASL) in real-time, converting them into text and equivalent images. It is of high priority due to its significant benefit in enhancing communication accessibility for the deaf and hard-of-hearing community.

2.4.1.2 Stimulus/Response Sequences

User Action: User performs an ASL gesture in view of the system's camera.
System Response: The system processes the webcam input, recognizes the gesture, and

displays the corresponding text and equivalent image.

2.4.1.3 Functional Requirements

- **REQ-1.1:** The system shall support real-time video input from a standard webcam or smartphone camera.
- **REQ-1.2:** The system shall use machine learning algorithms to recognize and interpret a predefined library of ASL gestures.
- **REQ-1.3:** The system shall convert recognized gestures into the corresponding text and equivalent image in real-time.
- **REQ-1.4:** The system shall provide visual feedback on the recognized gestures to the user through a user interface.
- **REQ-1.5:** In the case of unrecognized gestures, the system shall prompt the user to repeat the gesture or provide suggestions for what the user may have intended.
- **REQ-1.6:** The system shall be capable of operating in diverse lighting conditions, accommodating variations in user environment.

2.4.2 ISL Detection

2.4.2.1 Description and Priority

This feature enables the system to recognize and interpret gestures from Indian Sign Language (ISL) in real-time, converting them into text and equivalent images. It is of high priority due to its significant benefit in enhancing communication accessibility for the deaf and hard-of-hearing community.

2.4.2.2 Stimulus/Response Sequences

User Action: User performs an ISL gesture in view of the system's camera.
System Response: The system prompts the user to retry the gesture or offers suggestions for possible intended gestures.

2.4.2.3 Functional Requirements

- **REQ-2.1:** The system shall support real-time video input from a standard webcam or smartphone camera.
- **REQ-2.2:** The system shall use machine learning algorithms to recognize and interpret a predefined library of ISL gestures.
- **REQ-2.3:** The system shall convert recognized gestures into the corresponding text and/or synthesized equivalent image in real-time.
- **REQ-2.4:** The system shall provide visual feedback on the recognized gestures to the user through a user interface.
- **REQ-2.5:** In the case of unrecognized gestures, the system shall prompt the user to repeat the gesture or provide suggestions for what the user may have intended.
- **REQ-2.6:** The system shall be capable of operating in diverse lighting conditions, accommodating variations in user environment.

2.4.3 ASL to ISL Conversion

2.4.3.1 Description and Priority

This feature involves the development of a system capable of translating American Sign Language (ASL) gestures into Indian Sign Language (ISL) gestures and text. It would enable communication between individuals fluent in ASL and those fluent in ISL, bridging the gap between these two distinct sign languages.
Priority: Medium

2.4.3.2 Stimulus/Response Sequences

User Actions: User performs an ASL sign gesture.
System Response: Software captures the ASL gesture using the webcam.
User Actions: User confirms the captured ASL gesture.
System Response: Software processes the ASL gesture and begins translating it into ISL.
User Actions: User waits for the translation.

System Response: Software displays the translated ISL gesture and corresponding text on the screen.

User Actions: User selects the option to continue translating more ASL gestures.

User Actions: User exits the software.

System Response: Software closes the application.

2.4.3.3 Functional Requirements

- **REQ-3.1:** The software must capture ASL gestures accurately using the webcam.
- **REQ-3.2:** The software must translate captured ASL gestures into corresponding ISL gestures and text real-time with accuracy.
- **REQ-3.3:** The software must display the translated ISL gestures and text real-time with accuracy on the screen.
- **REQ-3.4:** The gestures must be unambiguous.
- **REQ-3.5:** Develop an algorithm for accurate translation between ASL and ISL.

2.4.4 ISL to ASL Conversion

2.4.4.1 Description and Priority

This feature involves the translation of sign language gestures between American Sign Language (ASL) and Indian Sign Language (ISL) in real-time.

Priority: High priority

2.4.4.2 Stimulus/Response Sequences

User Actions: User initiates the translation feature by selecting the desired sign language conversion (ASL to ISL or vice versa).

System Responses: System detects and interprets sign language gestures captured by the camera.

User Actions: User performs sign language gestures in front of the camera for translation.

System Responses: System translates the interpreted gestures into the target sign language (ASL to ISL or ISL to ASL) using machine learning algorithms.

User Actions: User confirms or corrects translated gestures as necessary.
System Responses: System displays the translated output,as text and sign language gesture,to the user.
User Actions: User interacts with the translated output, such as displaying text or playing back translated sign language gestures.

2.4.4.3 Functional Requirements

- **REQ-4.1:** The system must accurately interpret sign language gestures captured by the camera. If the system encounters ambiguous or unclear gestures, it should prompt the user for clarification or provide suggestions for alternative interpretations.
- **REQ-4.2:** The system must implement a translation algorithm capable of converting interpreted sign language gestures into the target sign language (ASL to ISL or ISL to ASL) in real-time.
- **REQ-4.3:** The system must provide real-time translation capabilities to maintain fluid communication between users.
- **REQ-4.4:** Develop an algorithm for accurate translation between ASL and ISL.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

- **Real-Time Translation Performance:** The system should provide translation results (from ASL to ISL and vice versa) within a maximum latency of 2 seconds from the completion of a sign gesture by the user.
- **Recognition Accuracy:** The sign language recognition engine must achieve a minimum accuracy rate of 95% under controlled lighting and background conditions. This high level of accuracy is essential for reducing misunderstandings and enhancing the effectiveness of the translation.

- **Uptime and Reliability:** The translation service should maintain an uptime of 99.9% to ensure availability to users across different time zones. This high level of reliability is crucial for a service that could be essential for users' daily communications.

- **Latency and Synchronization:** In scenarios where the translation involves sending data to and from a server, the total round-trip time should not exceed 500 milliseconds. This low latency is essential for maintaining the real-time nature of the translation service.

2.5.2 Software Quality Attributes

- The accuracy of translation from ASL to ISL should be high, with minimal errors in recognizing and interpreting signs.

- The system should be robust to variations in hand gestures, lighting conditions, background noise, and other environmental factors.

- The ease of use for both ASL speakers and ISL speakers is crucial.

- The ability of the system to run on different platforms and devices, such as desktop computers, mobile devices, or embedded systems, enhances its accessibility and availability.

Chapter 3

System Architecture and Design

3.1 System Overview

The project aims to develop a system capable of translating American Sign Language gestures to Indian Sign Language and vice versa in real time. This system will facilitate communication between individuals who use different sign languages, thereby bridging the communication gap.

3.2 Architectural Design

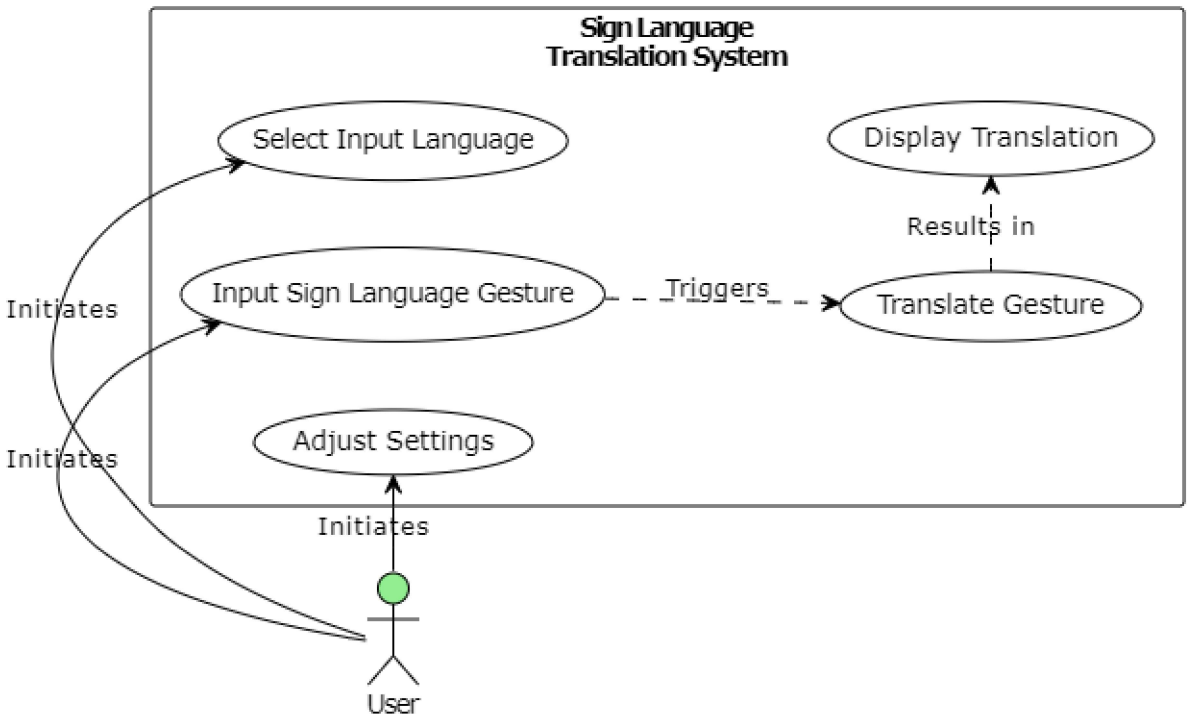


Figure 3.1: Use Case Diagram: Sign Language Translation System

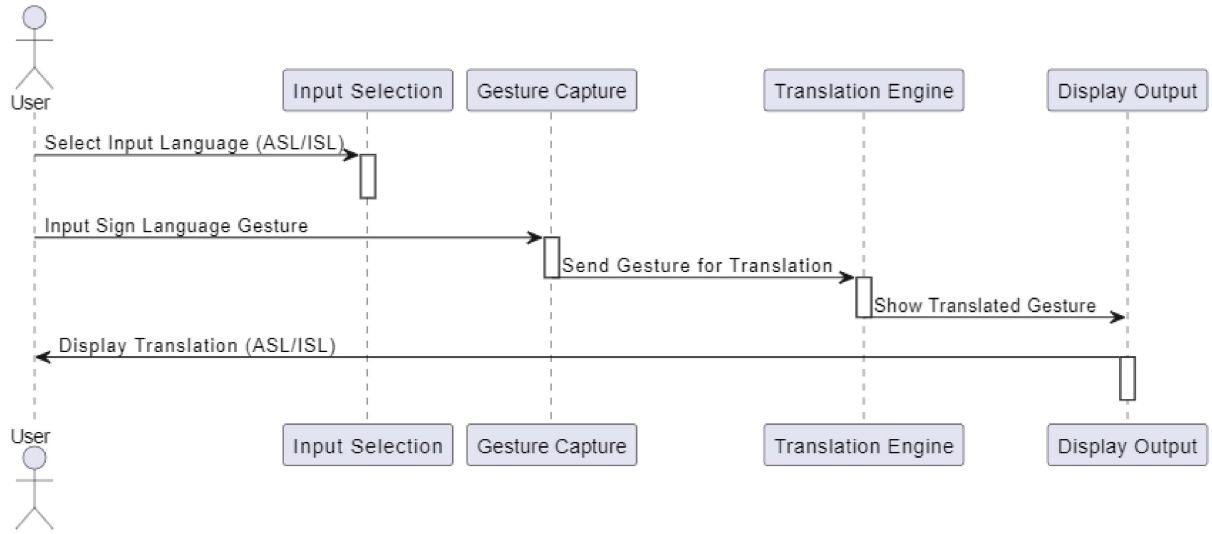


Figure 3.2: Sequence Diagram: Sign Language Translation System

3.3 Dataset identified

For the ambitious undertaking of translating between American Sign Language (ASL) and Indian Sign Language (ISL), a comprehensive and diverse dataset is paramount. This project leverages a meticulously curated dataset that encompasses a broad range of sign language gestures from both ASL and ISL. The dataset is designed to train and evaluate the translation system with high accuracy and efficiency, ensuring a seamless translation experience for users. Below, the properties of the dataset and its relevance to the project are outlined, along with highlights of sample subsets.

The datasets we will be using are:

ASL Dataset: This dataset contains alphabets and digits in ASL.

ISL Dataset: This dataset also contains alphabets and digits in ISL.

3.3.1 Dataset Properties

- **Size and Composition**: The dataset consists of images, with an equal distribution between ASL and ISL signs. Each entry is annotated with labels indicating the corresponding sign, ensuring a balanced representation of gestures.

- **Diversity and Variability**: To accommodate the variations in signing style, the

dataset includes signs performed by multiple signers of different ages, genders, and backgrounds. This diversity ensures the model's robustness and generalization across real-world scenarios.

- **Quality and Resolution**:Images are captured in high definition to preserve the nuances of each sign. This quality is crucial for accurately identifying and translating subtle hand gestures and movements.

3.3.2 Sample Subsets Highlight

- **ASL and ISL Alphabets and Digits Subset**: A subset featuring high-resolution images of the ASL and ISL alphabets and digits, showcasing the hand shapes for each letter and digit. This subset is crucial for foundational translation models focusing on character-by-character transcription.

- **Environmental Variability Subset**: This subset includes signs performed under various lighting conditions and backgrounds, testing the model's ability to adapt to different environments.

3.4 Proposed Methodology/Algorithms

- **Algorithm for ASL to ISL translation**:

```
def translate_asl_to_isl(frame):
    // Detect hands in the frame
    hands = detect_hands(frame)
    // Extract features from each hand
    features = extract_features(hands)
    // Classify gestures using SVM
    predicted_gestures = svm_classifier.predict(features)
    // Translate predicted gestures to ISL
    translated_text = translate_to_isl(predicted_gestures)
    return translated_text
```

- **Algorithm for ISL to ASL translation**:

```
def translate_isl_to_asl(frame):
    //Detect hands in the frame
    hands = detect_hands(frame)
    //Extract features from each hand
    features = extract_features(hands)
    //Classify gestures using SVM
    predicted_gestures = svm_classifier.predict(features)
    //Translate predicted gestures to ASL
    translated_text = translate_to_asl(predicted_gestures)
    return translated_text
```

3.5 User Interface Design

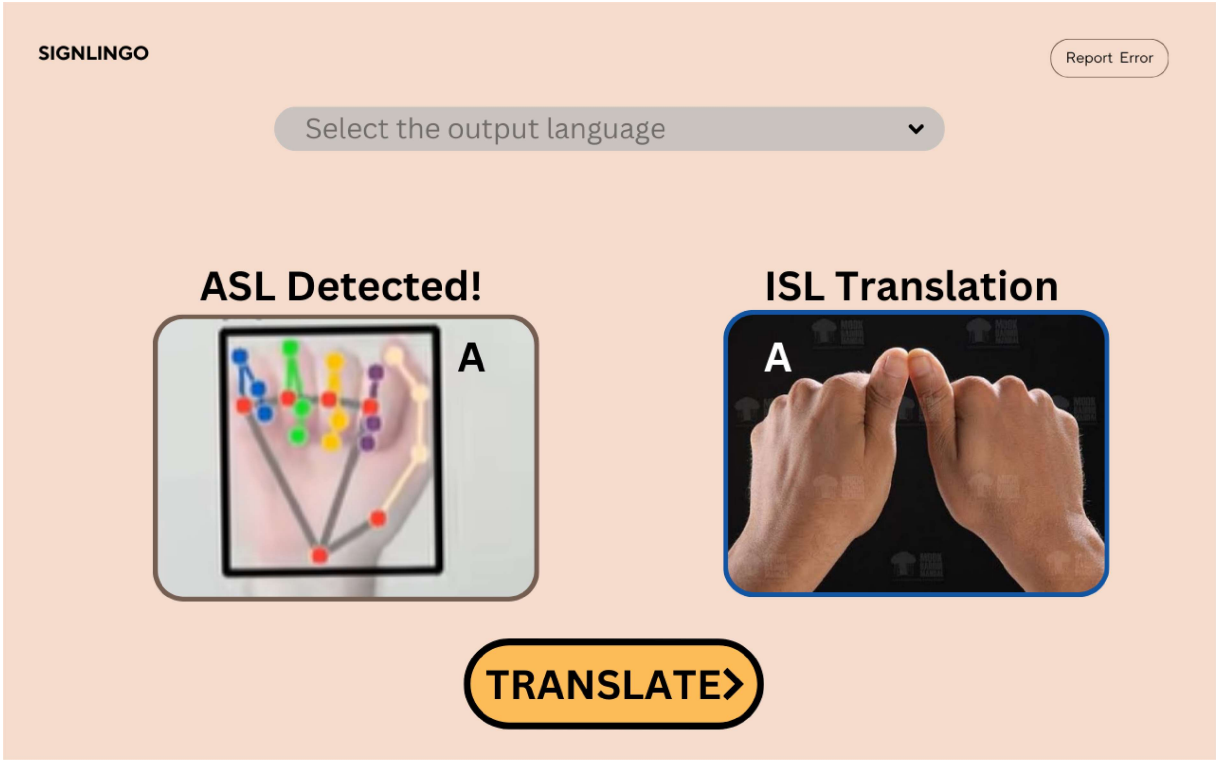


Figure 3.3: UI Wireframe: Sign Language Translation System (SignLingo)

3.6 Database Design

Detailed Database Design

- **Entities**
  - **Translations**: Records the translation requests and their results.
  - **Signs**: Contains the dictionary of sign language gestures and their corresponding interpretations.
- **Attributes**
  - **Translations**: Translation ID, User ID, Original Language, Target Language, Original Content, Translated Content, Timestamp.
  - **Signs**: Sign ID, Language, Gesture Description, Textual Interpretation, Video Reference.
- **Relationships**
  - Users can have multiple Translations.
  - Signs are referenced by Translations.

Database Schema

- **Translations**
  - `translation_id`: PRIMARY KEY
  - `user_id`: FOREIGN KEY
  - `original_language`: VARCHAR
  - `target_language`: VARCHAR
  - `original_content`: TEXT
  - `translated_content`: TEXT
  - `timestamp`: TIMESTAMP
- **Signs**
  - `sign_id`: PRIMARY KEY
  - `language`: VARCHAR
  - `gesture_description`: TEXT

- `textual_interpretation`: TEXT
- `video_reference`: VARCHAR

Choice of Database

For this project, a **relational database management system (RDBMS)** like PostgreSQL or MySQL is suitable due to the following reasons:

- **ACID Compliance**: Ensures reliable transaction processing which is crucial for maintaining the integrity of translation records and user feedback.
- **Relational Data Model**: Facilitates complex queries across different entities, which is essential for retrieving translations, user data, and feedback.
- **Scalability**: Both MySQL and PostgreSQL offer good scalability options, important as the number of users and translations grows.
- **Mature Ecosystem**: Both have mature tools and libraries for backup, monitoring, and high availability.
- **Community and Support**: Large communities and extensive documentation can aid in development and problem-solving.
- **Security**: Strong security features to protect sensitive user data.

3.7 Description of Implementation Strategies

Data Collection:

- Gather a diverse dataset of videos containing sign language gestures for both ASL and ISL.
- Ensure coverage of a wide range of signs and variations in hand movements and positions

Data Preprocessing:

- Preprocess sign language videos to extract individual frames.
- Segment videos into sequences representing individual signs.

- Normalize data and handle any noise or inconsistencies in video quality.

Feature Extraction:

- Extract relevant features from preprocessed frames capturing important aspects of sign language such as hand shapes and movement.
- Consider techniques like keypoint detection, motion tracking, or deep learning-based feature extraction methods.

Model Selection and Training:

- Choose appropriate machine learning or deep learning models such as CNNs, RNNs, or transformer models.
- Train selected models using extracted features and corresponding labels (ASL to ISL mappings and vice versa).
- Utilize techniques like transfer learning if pre-trained models are available and fine-tune models as necessary.

Evaluation:

- Evaluate trained models using appropriate metrics like accuracy, precision, recall, and F1-score.
- Validate models on separate test dataset to assess generalization performance.
- Iterate on model training and evaluation process to improve performance.

Deployment:

- Deploy trained models as a service or application accessible to users.
- Develop user-friendly interface for users to input sign language gestures (via image) and receive translations in real time.
- Ensure scalability, reliability, and efficiency of deployment infrastructure to handle multiple users.

Continuous Improvement:

- Collect feedback from users and incorporate it into the system to improve translation accuracy and user experience.
- Monitor performance of deployed models and update them periodically with new data and techniques to adapt to evolving sign language patterns and variations.

3.8 Module Division

1. **Data Acquisition Module:** This module captures sign language data. It can employ:
    - Cameras for video recordings.
    - This module will be handled by Aarathi Nair
2. **Pre-processing Module:** This module prepares the captured data for further processing. It performs tasks like:
    - Video segmentation to isolate the signer from the background.
    - Hand segmentation to focus on the hands and their movements.
    - Normalization to ensure consistency in data format and size.
    - This module will be handled by Aarathi Nair
3. **Feature Extraction Module:** This module extracts the essential features from the pre-processed data that represent the signs. It involves:
    - Hand pose estimation to identify the handshapes and their orientation.
    - Movement tracking to capture the motion patterns of the hands.
    - This module will be handled by Anitta Mariya Shaji
4. **Sign Recognition Module:** This module recognizes the individual signs based on the extracted features. It often uses:
    - Machine learning models trained on large datasets of ASL and ISL signs.

- Deep learning techniques like Convolutional Neural Networks (CNNs) for robust recognition.
- This module will be handled by Anushri Dilip

5. **Language Translation Module:** This module translates the recognized signs into the target language (ISL or ASL). It might involve:
   - Sign language grammar analysis to understand the structure and order of signs.
   - Machine translation techniques specifically designed for sign languages (still under development).
   - This module will be handled by Anjala Binu
6. **Output Generation Module:** This module presents the translated text or signs to the user. It can generate:
   - Textual output displaying the translated alphabet or digit in ISL or ASL.
   - Visual output using images to depict the translated ISL or ASL signs.
   - This module will be handled by Anitta Mariya Shaji

3.9 Work Schedule - Gantt Chart

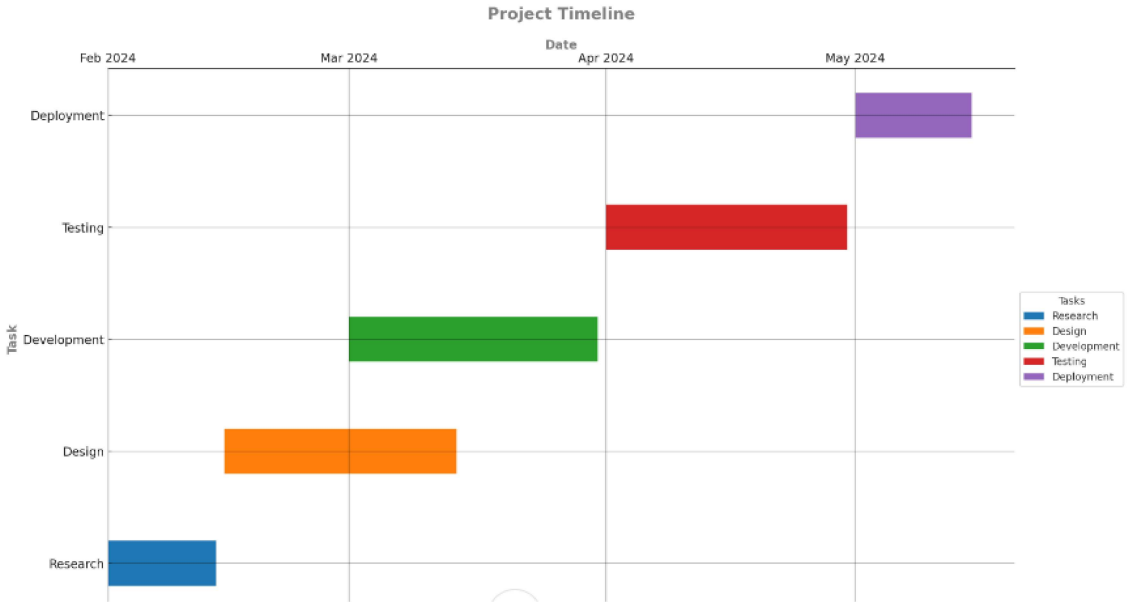


Figure 3.4: Gantt Chart

Chapter 4

Results and Discussions

4.1 Overview

Our project focused on translating American Sign Language (ASL) to Indian Sign Language (ISL) and vice versa, using sign gestures as input and providing equivalent gestures and text as output. The system has been successfully developed and tested for its core functionalities, demonstrating the potential to facilitate communication between ASL and ISL users. Initial qualitative testing indicates that the system can recognize and translate a variety of gestures from both ASL and ISL.The system is user-friendly and effective in bridging the communication gap between different sign language users.

4.2 Testing

Figure 4.1: Title page

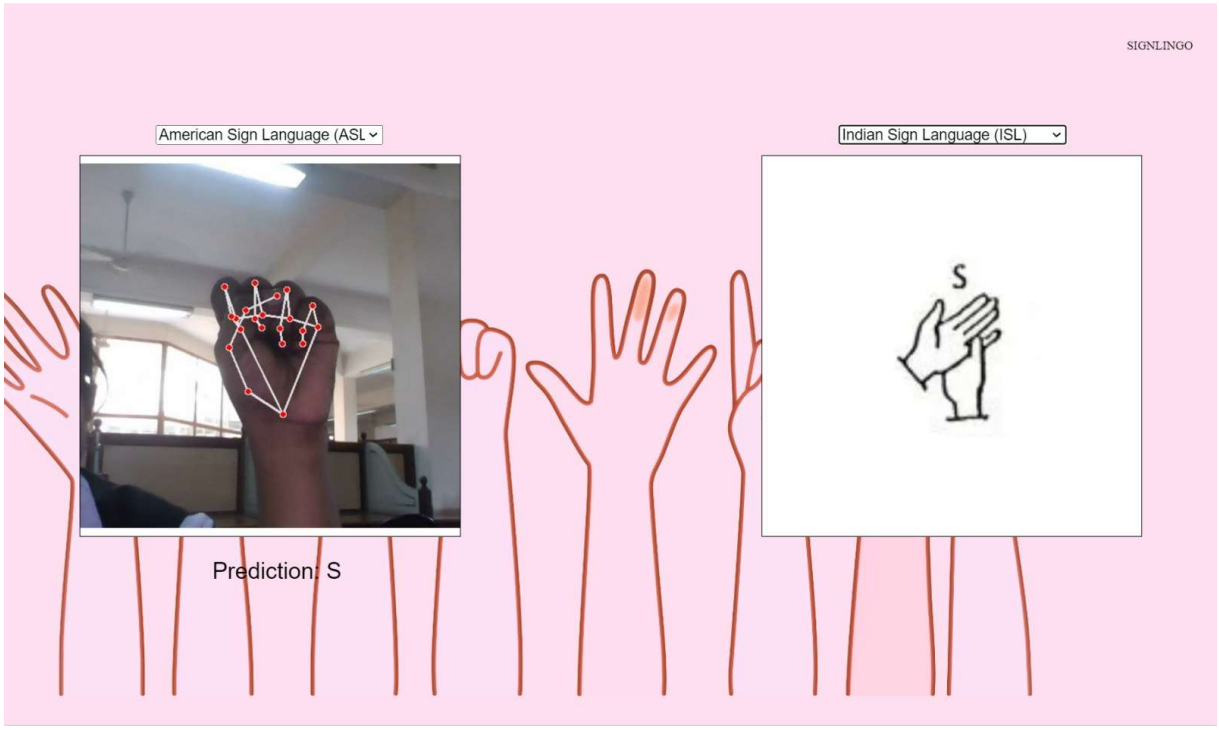


Figure 4.2: Translation for letter 'S'

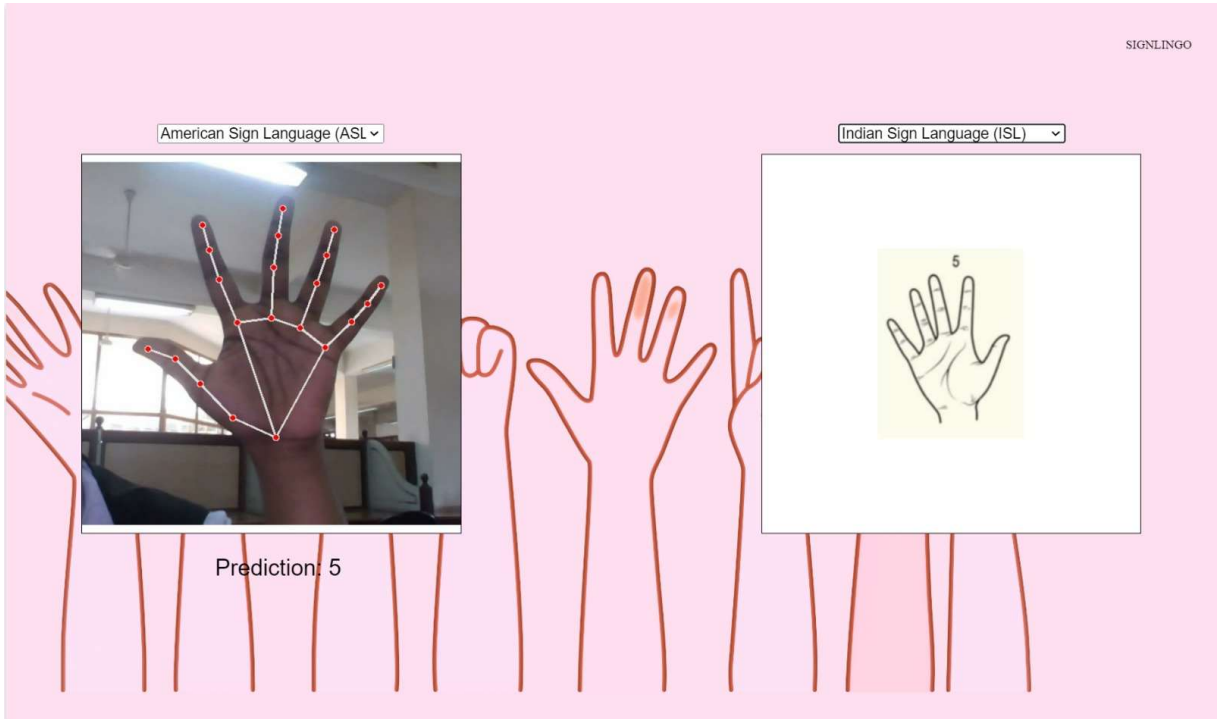


Figure 4.3: Translation for number '5'

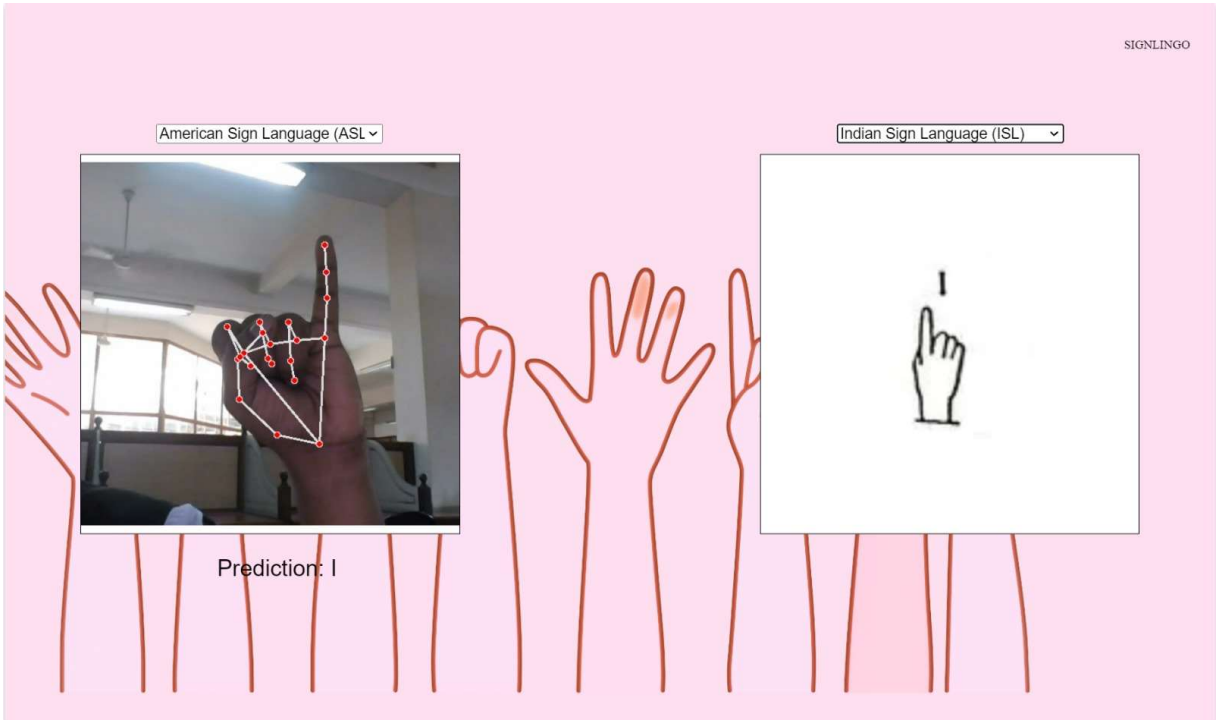


Figure 4.4: Translation for letter 'I'

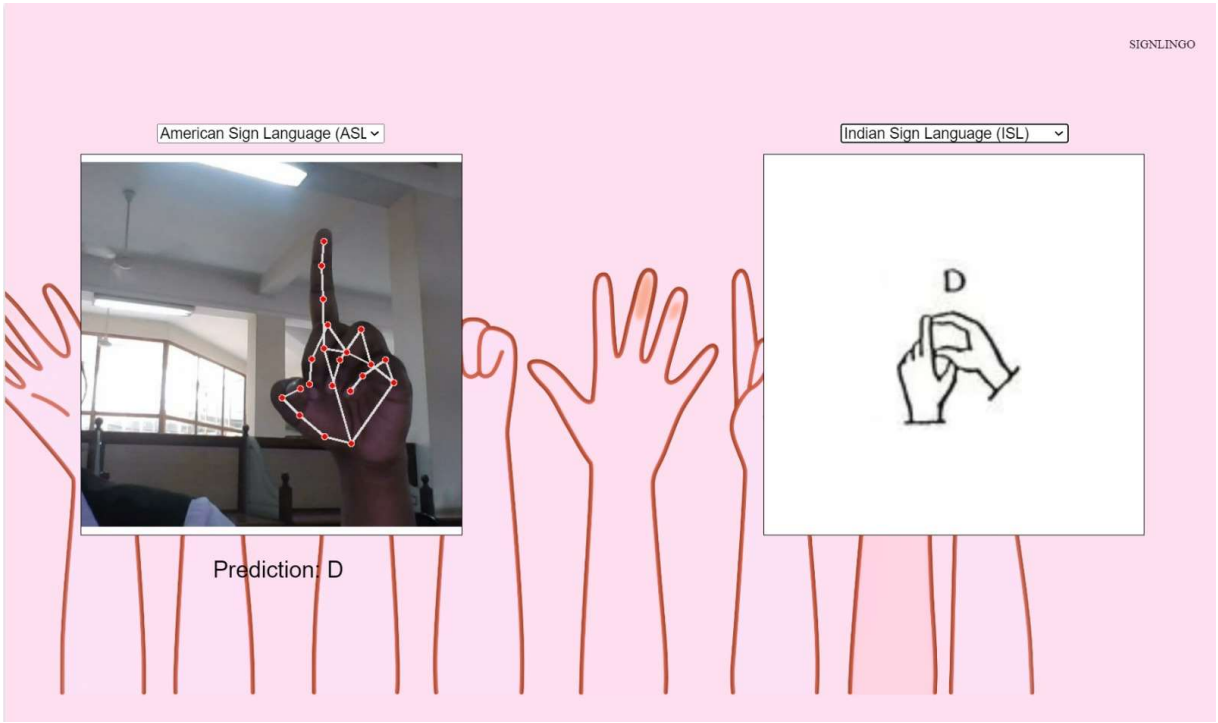


Figure 4.5: Translation for letter 'D'

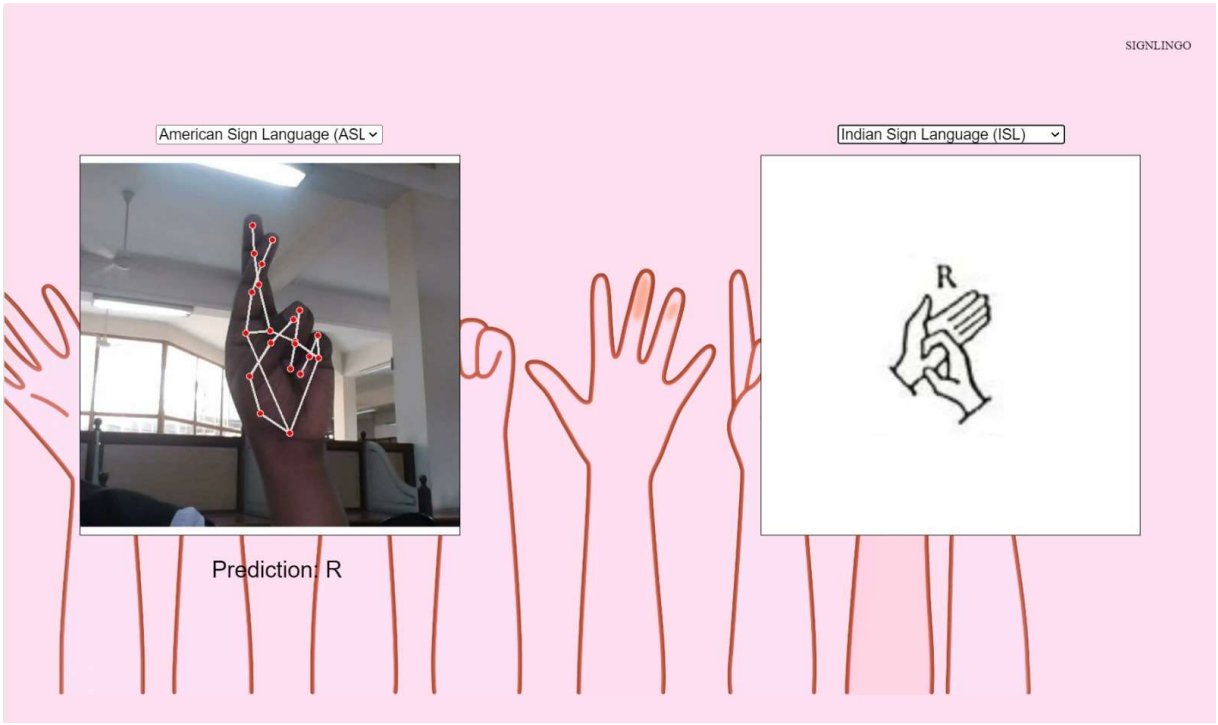


Figure 4.6: Translation for letter 'R'

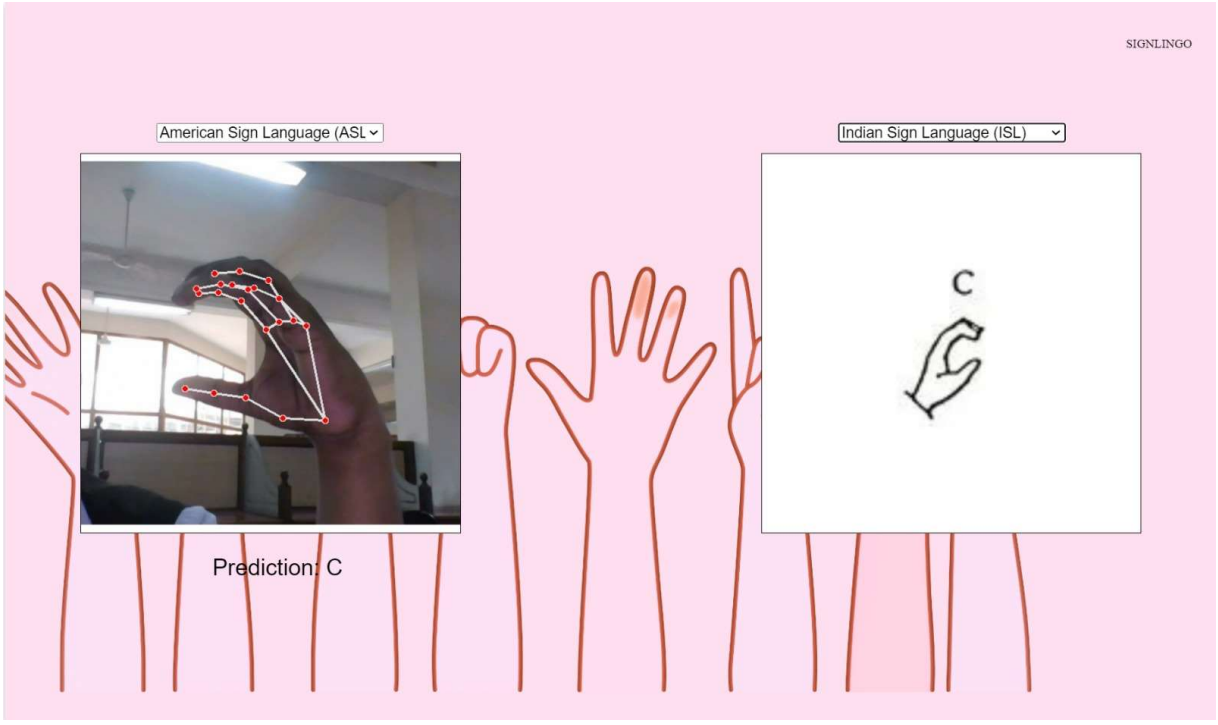


Figure 4.7: Translation for letter 'C'

4.3 Discussion

Our sign language translation system has yielded notable outcomes, bridging American Sign Language (ASL) and Indian Sign Language (ISL). Our project's intersection of technology and accessibility aims to empower individuals with hearing impairments. Looking ahead, avenues for enhancement include adaptive learning algorithms, gesture recognition refinement, and expanded language support. Collaborations with linguists and sign language experts enrich our understanding, aligning with evolving standards and user expectations. In essence, our commitment to inclusivity remains steadfast, propelled by iterative improvements and a user-centric approach.

Chapter 5

Conclusion

5.1 Conclusion

The Sign Language Translation System is a software solution that translates sign language gestures between American Sign Language (ASL) and Indian Sign Language (ISL), providing both gestural and textual translations.This innovative software captures ASL and ISL gestures through a user-friendly interface, incorporates advanced sign language detection, accurately translating them into the target sign language and accompanying text. It supports real-time translation, enhancing communication accessibility for ASL and ISL users with both visual and textual outputs.

The system presented offers a comprehensive solution for bridging communication barriers between Indian Sign Language (ISL) and American Sign Language (ASL) within a single web-based platform. By integrating real-time video input, users can initiate translation sessions directly on the webpage, allowing for seamless interaction and communication between individuals who use these sign language gestures. Through well-built user interfaces, users can select the input and output languages, facilitating efficient and accessible translation services. This system not only enhances accessibility and inclusivity for individuals who rely on sign languages for communication but also fosters cultural exchange and understanding between communities using different sign languages. By providing a centralized platform for sign language translation, this system promotes cross-cultural collaboration and empowers individuals to engage in meaningful and effective communication regardless of geographical or linguistic barriers. With its user-friendly design and effective functionality, the Sign Language Translation System displays the potential of technology to facilitate communication and promote inclusivity in diverse communities.

5.2 Future Scope

The Sign Language Translation System will introduce real-time collaboration through features like multi-user translation sessions and group video calls, facilitating seamless teamwork and communication across different sign languages and geographical locations.A major future scope of the project involves expanding language support to include a wider range of sign languages from diverse regions worldwide. This expansion will cater to the diverse needs of users globally, fostering inclusivity and accessibility in communication.Feedback mechanisms that will collect user input to enhance translation accuracy, usability, and features, ensuring continuous improvement based on community feedback and suggestions.By utilizing cloud-based services, user preferences, translation history, and learning materials will be securely stored and accessible across devices and platforms, enhancing convenience and flexibility.Further it can also be extended to educational sector for providing educational resources and interactive tutorials to assist users in learning sign languages.

References

[1] . Natarajan, E. Rajalakshmi, R. Elakkiya, Ketan Kotecha, Ajith Abraham, Lubna Abdelkareim Gabralla, V. Subramaniyaswamy, 2022, "Development of an End-to-End Deep Learning Framework for Sign Language Recognition, Translation, and Video Generation"

[2] . Sahoo, G. Mishra, K. Ravulakollu, 2014, "Sign Language Recognition: State of the Art"

[3] . Papatsimouli, K. Kollias, L. Lazaridis, G. Maraslidis, H. Michailidis, P. Sarigiannidis and G. Fragulis, 2022, "Real Time Sign Language Translation Systems: A review study"

[4] iming He, 2019, "Research of a Sign Language Translation System Based on Deep Learning"

[5] .Grover, R. Aggarwal, D. Sharma, P. Gupta, 2021, "Sign Language Translation Systems for Hearing/Speech Impaired People: A Review"

[6] . Abraham, A. Nayak, A. Iqbal, 2019, "Real-Time Translation of Indian Sign Language using LSTM"

Appendix A: Presentation

Sign Language Translation System

Ms Anita John
Asst Professor

Aarathi Nair- U2003002
Anitta Mariya Shaji- U2103034
Anjala Binu- U2103035
Anushri Dilip- U2103043

Contents

Introduction

- Sign language translation systems are vital for tackling accessibility challenges and fostering inclusivity.
- They effortlessly translate between American Sign Language (ASL) and Indian Sign Language (ISL), facilitating smooth communication across diverse sign language communities.
- This technology aims to bridge communication gaps and empower users in social, educational, and professional settings.
- These systems provide instant access to essential information, educational materials, and job opportunities, which are often inaccessible due to limited support for sign language communication.

Problem Definition

The aim is to create a proficient sign language translation system that seamlessly translates between American Sign Language (ASL) and Indian Sign Language (ISL), enabling effective communication within diverse sign language communities.

Objectives

Objective 1: Data Collection and Annotation

- Gather a diverse dataset of ASL and ISL gestures as images covering various expressions and vocabulary.
- Annotate the collected images with linguistic information to train the translation model effectively.

Objective 2: Model Development and Training

- Develop a deep learning architecture suitable for sign language translation, considering the nuances of ASL and ISL.
- Train the model using the annotated dataset to learn the mapping between ASL and ISL gestures, focusing on accuracy and efficiency.

Objectives

Objective 3: Evaluation and Refinement

- Assess the model's performance using standard metrics and real-world scenarios.
- Refine the model based on evaluation results and user feedback to improve translation accuracy and fluency.

Objective 4: Deployment and User Interface

- Integrate the translation system into an accessible application for deaf individuals and sign language users.
- Design a user-friendly interface allowing seamless input and output of ASL or ISL gestures for real-time translation.

Scope and Relevance

SCOPE

- Translation Between ASL and ISL: Develops an efficient system to translate gestures and expressions between American Sign Language and Indian Sign Language.
- Real-time Translation: Utilizes advanced computer vision and machine learning techniques to ensure accurate and real-time translation.
- User-Friendly Interface: Provides an intuitive and accessible interface for users to facilitate seamless communication.

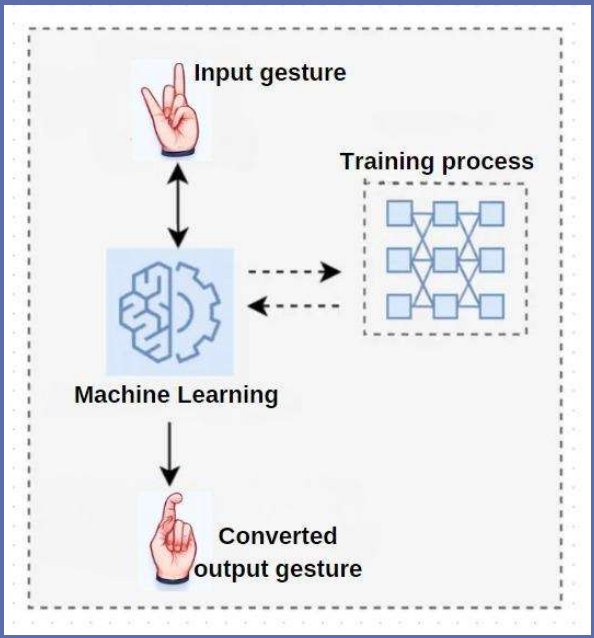
RELEVANCE

- Bridging Communication Gaps: Enhances communication between ASL and ISL users, fostering better understanding and interaction.
- Inclusivity: Promotes inclusivity for the deaf and hard-of-hearing communities by breaking down language barriers.
- Global Connectivity: Supports global collaboration and communication among sign language users from different regions.
- Educational Tool: Assists in learning and teaching ASL and ISL, benefitting educators and students alike.

System Design-System Overview

- Gesture Recognition Technology: Utilizes state-of-the-art technology to detect and interpret sign language gestures via cameras or sensing devices.
- Textual Translation: Alongside gestural translation, the system provides a textual output of the translated signs, enhancing accessibility and understanding.
- Real-time Translation: Offers immediate translation to ensure smooth and natural conversation flow between ASL and ISL users.
- Security and Privacy: Incorporates strong measures to protect user data and maintain confidentiality.
- Inclusivity and Accessibility: Aims to make digital communication and public services more accessible for the deaf and hard of hearing community, promoting inclusivity across linguistic and cultural boundaries.

System Design-Architectural Design



System Design - Modules

1. Data Acquisition Module: This module captures sign language data. It can employ:
    - Cameras for video capture.

2. Pre-processing Module: This module prepares the captured data for further processing. It performs tasks like:
    - Video segmentation to isolate the signer from the background.
    - Hand segmentation to focus on the hands and their movements.
    - Normalization to ensure consistency in data format and size.

System Design - Modules

3. Feature Extraction Module: This module extracts the essential features from the pre-processed data that represent the signs. It involves:
    - Hand pose estimation to identify the handshapes and their orientation.
    - Movement tracking to capture the motion patterns of the hands.

4. Sign Recognition Module: This module recognizes the individual signs based on the extracted features. It often uses:
    - Video segmentation to isolate the signer from the background.
    - Hand segmentation to focus on the hands and their movements.
    - Normalization to ensure consistency in data format and size.

System Design - Modules

5. Language Translation Module: This module translates the recognized signs into the target language (ISL or ASL). It might involve:
    - Machine translation techniques specifically designed for sign languages (still under development).

6. Output Generation Module: This module presents the translated text or signs to the user. It can generate:
    - Textual output displaying the translated sentence in ISL or ASL
    - Visual output using images to depict the translated ISL or ASL signs.

System Design - Algorithms

- Algorithm for ASL to ISL translation:

```
def translate_asl_to_isl(frame):
// Detect hands in the frame
hands = detect_hands(frame)
// Extract features from each hand
features = extract_features(hands)
// Classify gestures using SVM
predicted_gestures = svm_classifier.predict(features)
// Translate predicted gestures to ISL
translated_text = translate_to_isl(predicted_gestures)
return translated_text
```

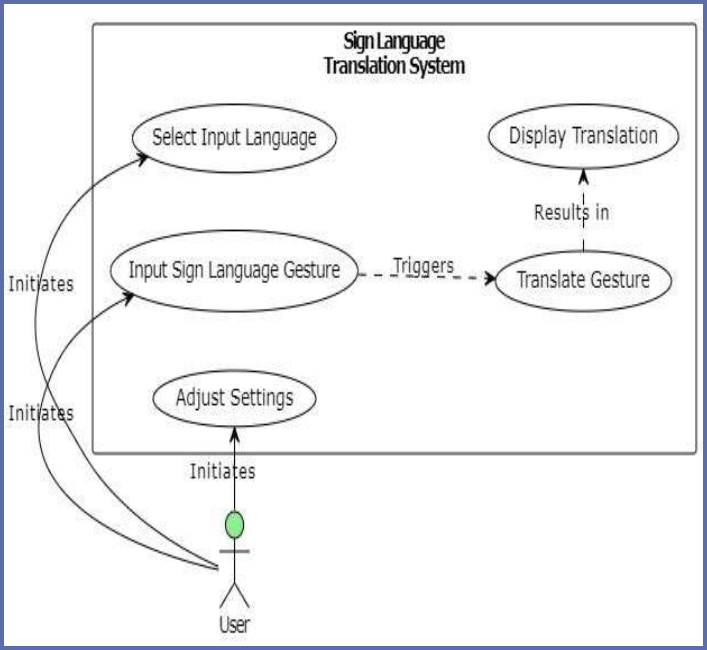

System Design - Algorithms

- Algorithm for ISL to ASL translation:

```
def translate_isl_to_asl(frame):
//Detect hands in the frame
hands = detect_hands(frame)
//Extract features from each hand
features = extract_features(hands)
//Classify gestures using SVM
predicted_gestures = svm_classifier.predict(features)
//Translate predicted gestures to ASL
translated_text = translate_to_asl(predicted_gestures)
return translated_text
```

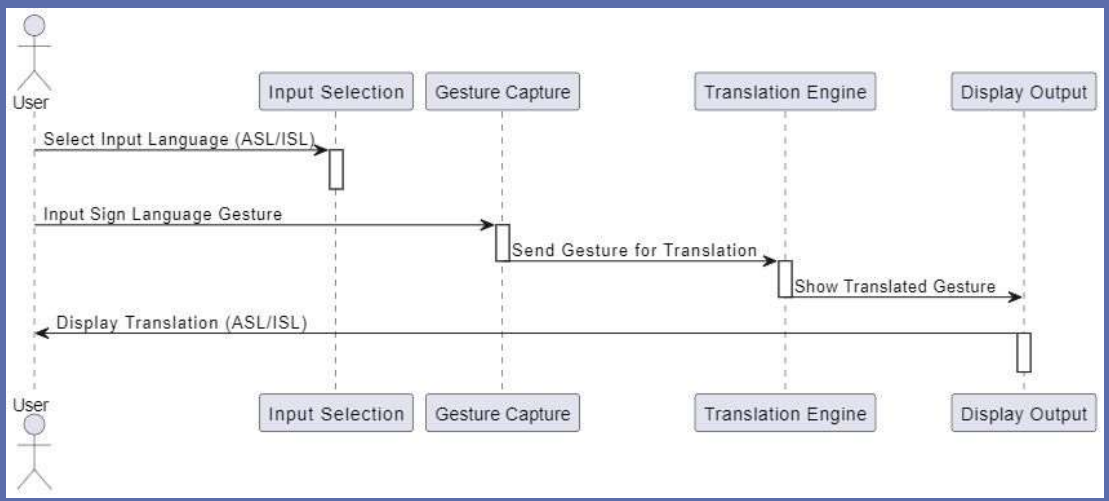
System Design - Design Models

USE-CASE DIAGRAM



System Design - Design Models

SEQUENCE DIAGRAM

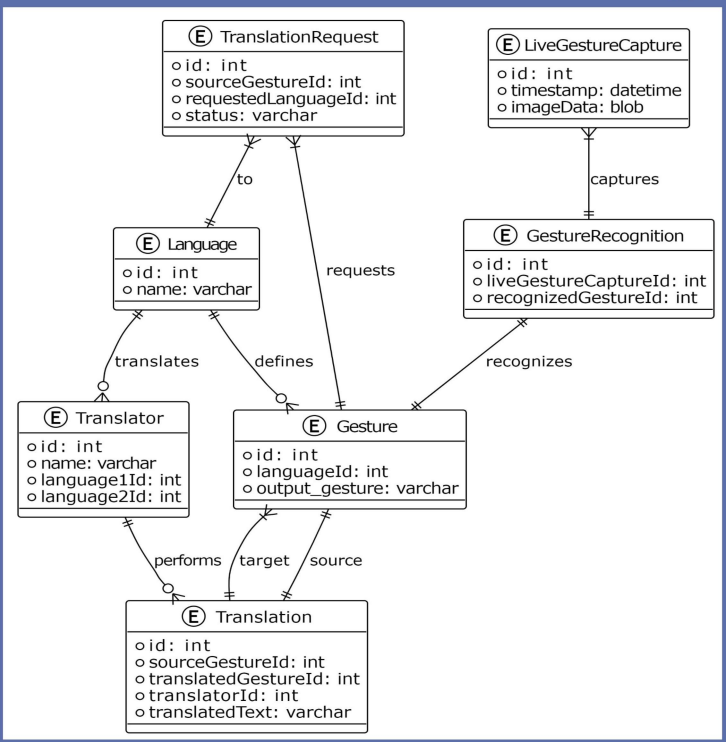


Datasets

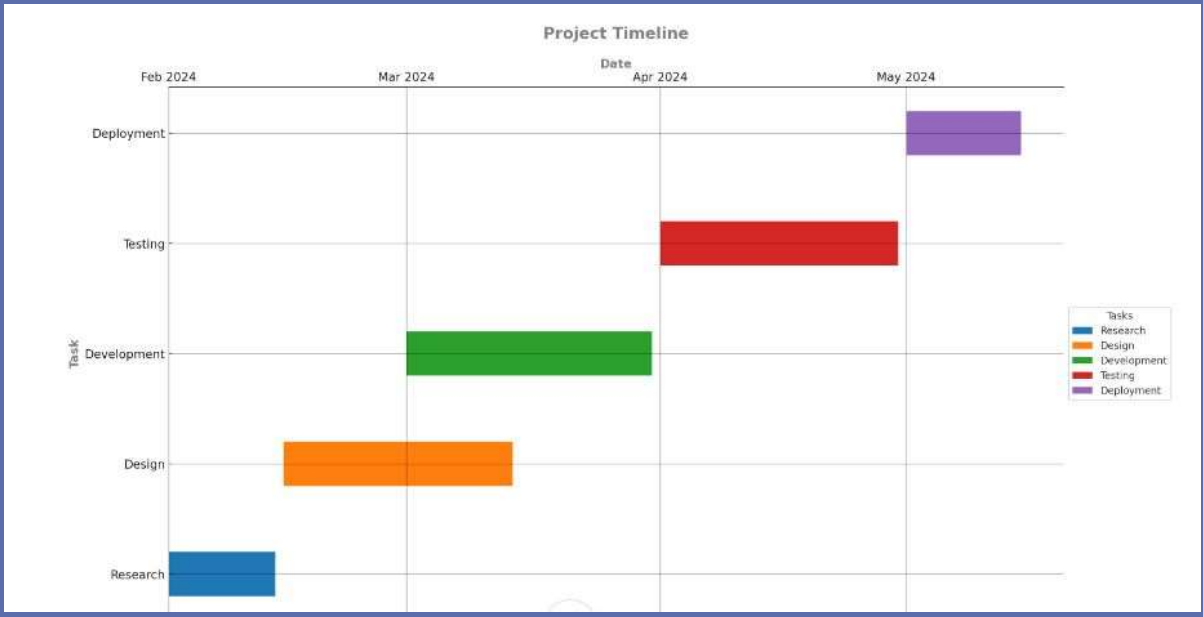
- ASL Datasets
  - Sources: kaggle, huggingface
  - Dataset contains alphabets(A-Z), digits(0-9) and a few common words like thank you, nice, hello, etc. in ASL

- ISL Datasets
  - Sources: kaggle, huggingface
  - Dataset contains alphabets(A-Z), digits(0-9) and a few common words like thank you, nice, hello, etc. in ISL

Database Schema Design



Work Division - Gantt Chart



Software/ Hardware Requirements

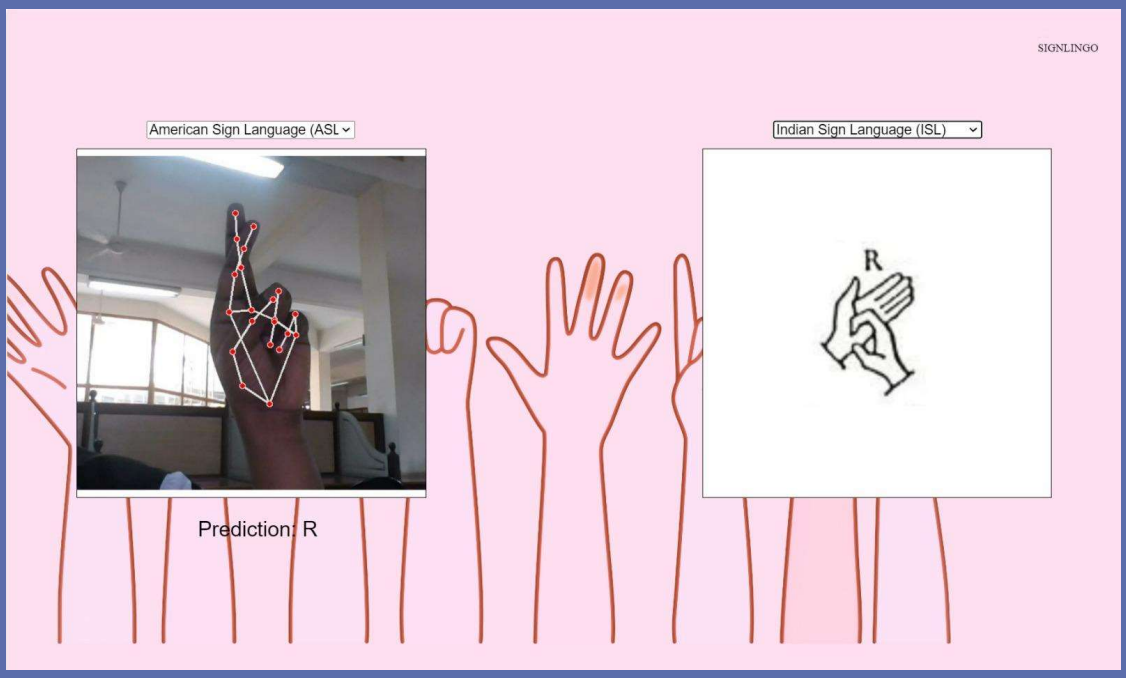
- **Software:**
  - Flask
  - OpenCV
  - JavaScript and HTML/CSS
  - TensorFlow/PyTorch

- **Hardware:**
  - 8GB RAM or higher
  - Integrated graphics or dedicated GPU for smoother video processing
  - High-definition webcam or camera with at least 720p resolution

Results

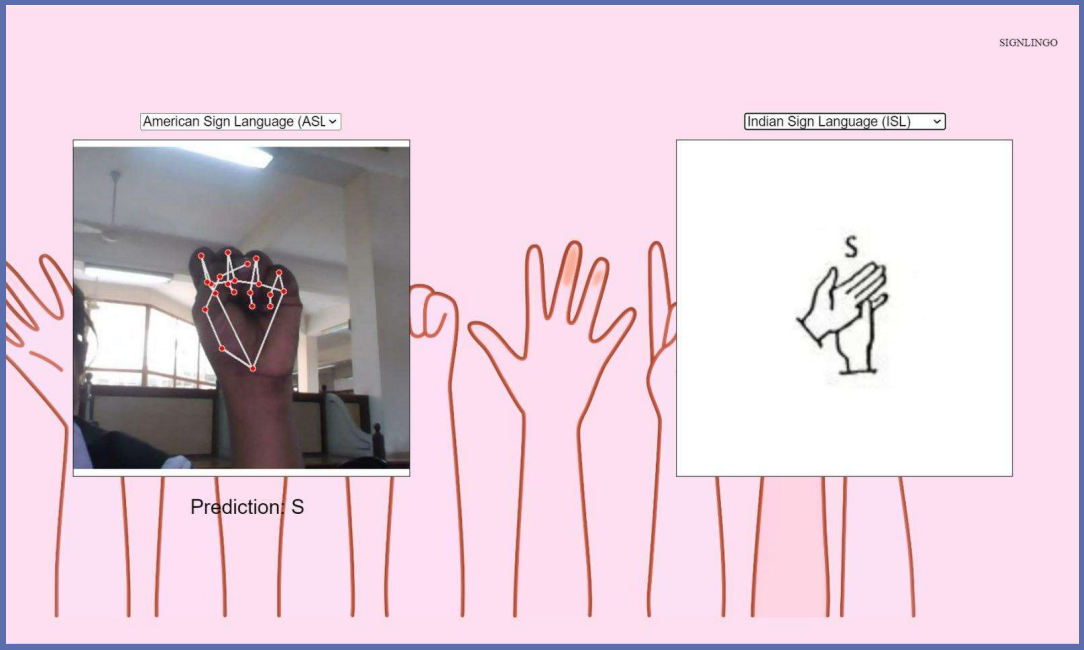
Title page

Results



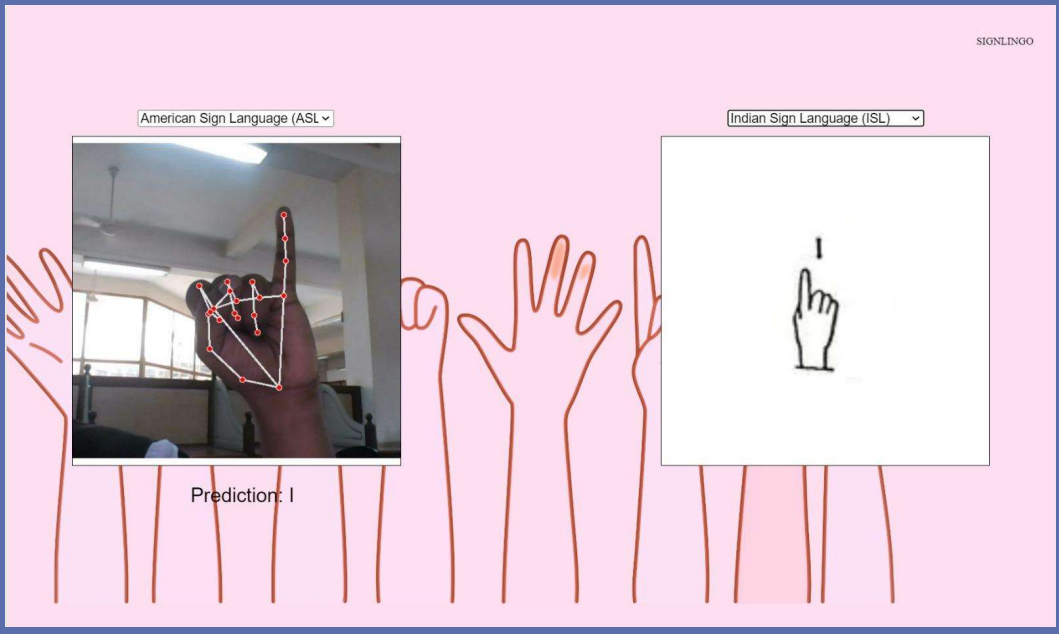
Translation for letter R

Results



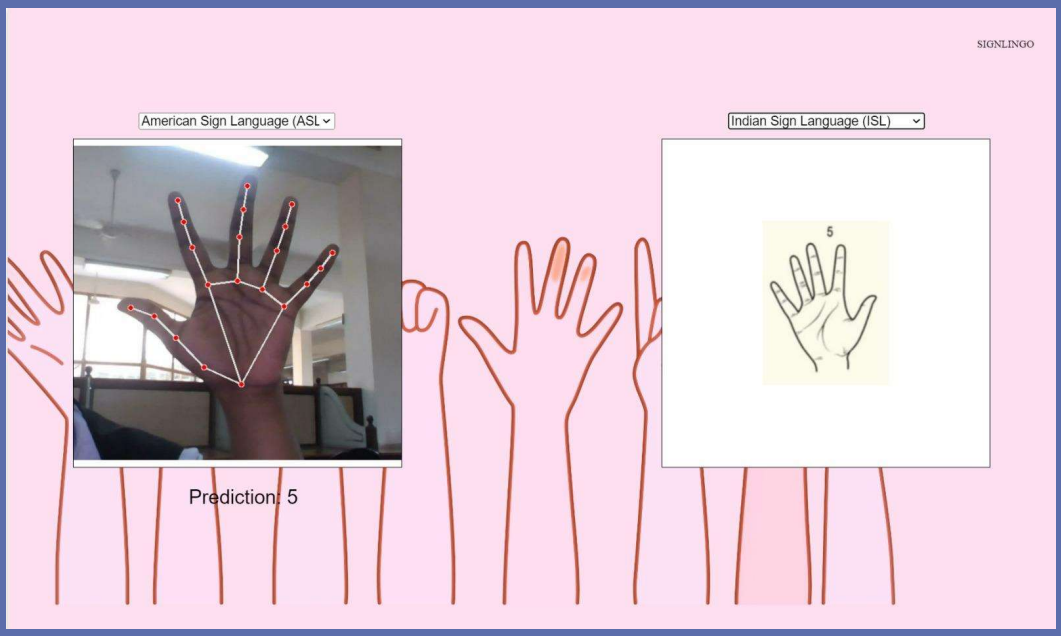
Translation for letter S

Results



Translation for letter I

Results



Translation for number 5

Conclusion

- A software solution that translates sign language gestures between ASL and ISL, delivering gestural and textual translations.

- This innovative software captures ASL and ISL gestures through a user-friendly interface,advanced sign language detection, accurately translating them into the target sign language and accompanying text. It supports real-time translation, enhancing communication accessibility for ASL and ISL users by breaking down language barriers with both visual and textual outputs.

Future Enhancements

- Support for Additional Sign Languages: Expand the system to include other sign languages such as British Sign Language (BSL), Japanese Sign Language (JSL).
- Enhanced Gesture Library: Continuously update the gesture database to include more nuanced and regional variations of ASL and ISL.
- Speech-to-Sign Translation: Integrate voice recognition to provide real-time translation of spoken language to sign language.
- Educational Sector: Providing tutorials and interactive sessions to the targeted communities.

References

- Papatsimouli, M., Kollias, K.F., Lazaridis, L., Maraslidis, G., Michailidis, H., Sarigiannidis, P. and Fragulis, G.F., 2022, June. Real Time Sign Language Translation Systems: A review study. In 2022 11th International Conference on Modern Circuits and Systems Technologies (MOCAST) (pp. 1-4). IEEE.
- Abraham, E., Nayak, A. and Iqbal, A., 2019, October. Real-time translation of Indian sign language using LSTM. In 2019 global conference for advancement in technology (GCAT) (pp. 1-5). IEEE.
- Grover, Y., Aggarwal, R., Sharma, D. and Gupta, P.K., 2021, February. Sign language translation systems for hearing/speech impaired people: a review. In 2021 International Conference on Innovative Practices in Technology and Management (ICIPTM) (pp. 10-14). IEEE.

References

- Natarajan, B., Rajalakshmi, E., Elakkiya, R., Kotecha, K., Abraham, A., Gabralla, L.A. and Subramaniyaswamy, V., 2022. Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation. *IEEE Access*, *10*, pp.104358-104374.

- Sahoo, A.K., Mishra, G.S. and Ravulakollu, K.K., 2014. Sign language recognition: State of the art. *ARPN Journal of Engineering and Applied Sciences*, *9*(2), pp.116-134.

- He, S., 2019, October. Research of a sign language translation system based on deep learning. In *2019 International conference on artificial intelligence and advanced manufacturing (AIAM)* (pp. 392-396). IEEE.

Appendix B: Vision, Mission, Programme Outcomes and Course Outcomes

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

RAJAGIRI SCHOOL OF ENGINEERING & TECHNOLOGY (AUTONOMOUS)

RAJAGIRI VALLEY, KAKKANAD, KOCHI, 682039

(Affiliated to APJ Abdul Kalam Technological University)

Vision, Mission, Programme Outcomes and Course Outcomes

Institute Vision

To evolve into a premier technological institution, moulding eminent professionals with creative minds, innovative ideas and sound practical skill, and to shape a future where technology works for the enrichment of mankind.

Institute Mission

To impart state-of-the-art knowledge to individuals in various technological disciplines and to inculcate in them a high degree of social consciousness and human values, thereby enabling them to face the challenges of life with courage and conviction.

Department Vision

To become a centre of excellence in Computer Science and Engineering, moulding professionals catering to the research and professional needs of national and international organizations.

Department Mission

To inspire and nurture students, with up-to-date knowledge in Computer Science and Engineering, ethics, team spirit, leadership abilities, innovation and creativity to come out with solutions meeting societal needs.

Programme Outcomes (PO)

Engineering Graduates will be able to:

1. Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

3. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

4. Conduct investigations of complex problems: Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

5. Modern Tool Usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

6. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

7. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

8. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

9. Individual and Team work: Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.

10. Communication: Communicate effectively with the engineering community and with society at large. Be able to comprehend and write effective reports documentation. Make effective presentations, and give and receive clear instructions.

11. Project management and finance: Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments.

12. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change.

Programme Specific Outcomes (PSO)

A graduate of the Computer Science and Engineering Program will demonstrate:

PSO1: Computer Science Specific Skills

The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas by understanding the core principles and concepts of computer science and thereby engage in national grand challenges.

PSO2: Programming and Software Development Skills

The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry.

PSO3: Professional Skills

The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur.

Course Outcomes

After the completion of the course the student will be able to:

CO1:

Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)

CO2:

Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)

CO3:

Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)

CO4:

Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)

CO5:

Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)

Appendix C: CO-PO-PSO Mapping

COURSE OUTCOMES:

After completion of the course the student will be able to

SL. NO	DESCRIPTION	Blooms' Taxonomy Level
CO1	Identify technically and economically feasible problems (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO2	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO3	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions of minimal complexity by using modern tools & advanced programming techniques (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO4	Prepare technical report and deliver presentation (Cognitive Knowledge Level: Apply)	Level 3: Apply
CO5	Apply engineering and management principles to achieve the goal of the project (Cognitive Knowledge Level: Apply)	Level 3: Apply

CO-PO AND CO-PSO MAPPING

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO3
CO1	3	3	3	3		2	2	3	2	2	2	3	2	2	2
CO2	3	3	3	3	3	2		3	2	3	2	3	2	2	2
CO3	3	3	3	3	3	2	2	3	2	2	2	3			2
CO4	2	3	2	2	2			3	3	3	2	3	2	2	2
CO5	3	3	3	2	2	2	2	3	2		2	3	2	2	2

3/2/1: high/medium/low

JUSTIFICATIONS FOR CO-PO MAPPING

MAPPING	LOW/ MEDIUM/ HIGH	JUSTIFICATION
100003/CS622T.1-PO1	**HIGH**	Identify technically and economically feasible problems by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.1-PO2	**HIGH**	Identify technically and economically feasible problems by analysing complex engineering problems reaching substantiated conclusions using first principles of mathematics.
100003/CS622T.1-PO3	**HIGH**	Design solutions for complex engineering problems by identifying technically and economically feasible problems.
100003/CS622T.1-PO4	**HIGH**	Identify technically and economically feasible problems by analysis and interpretation of data.
100003/CS622T.1-PO6	**MEDIUM**	Responsibilities relevant to the professional engineering practice by identifying the problem.
100003/CS622T.1-PO7	**MEDIUM**	Identify technically and economically feasible problems by understanding the impact of the professional engineering solutions.
100003/CS622T.1-PO8	**HIGH**	Apply ethical principles and commit to professional ethics to identify technically and economically feasible problems.
100003/CS622T.1-PO9	**MEDIUM**	Identify technically and economically feasible problems by working as a team.
100003/CS622T.1-PO10	**MEDIUM**	Communicate effectively with the engineering community by identifying technically and economically feasible problems.
100003/CS622T.1-P011	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles by selecting the technically and economically feasible problems.
100003/CS622T.1-PO12	**HIGH**	Identify technically and economically feasible problems for long term learning.
100003/CS622T.1-PSO1	**MEDIUM**	Ability to identify, analyze and design solutions to identify technically and economically feasible problems.
100003/CS622T.1-PSO2	**MEDIUM**	By designing algorithms and applying standard practices in software project development and Identifying technically and economically feasible problems.
100003/CS622T.1-PSO3	**MEDIUM**	Fundamentals of computer science in competitive research can be applied to Identify technically and economically feasible problems.
100003/CS622T.2-PO1	**HIGH**	Identify and survey the relevant by applying the knowledge of mathematics, science, engineering fundamentals.

100003/CS622T.2-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems get familiarized with software development processes.
100003/CS622T.2-PO3	**HIGH**	Design solutions for complex engineering problems and design based on the relevant literature.
100003/CS622T.2-PO4	**HIGH**	Use research-based knowledge including design of experiments based on relevant literature.
100003/CS622T.2-PO5	**HIGH**	Identify and survey the relevant literature for getting exposed to related solutions and get familiarized with software development processes by using modern tools.
100003/CS622T.2-PO6	**MEDIUM**	Create, select, and apply appropriate techniques, resources, by identifying and surveying the relevant literature.
100003/CS622T.2-PO8	**HIGH**	Apply ethical principles and commit to professional ethics based on the relevant literature.
100003/CS622T.2-PO9	**MEDIUM**	Identify and survey the relevant literature as a team.
100003/CS622T.2-PO10	**HIGH**	Identify and survey the relevant literature for a good communication to the engineering fraternity.
100003/CS622T.2-PO11	**MEDIUM**	Identify and survey the relevant literature to demonstrate knowledge and understanding of engineering and management principles.
100003/CS622T.2-PO12	**HIGH**	Identify and survey the relevant literature for independent and lifelong learning.
100003/CS622T.2-PSO1	**MEDIUM**	Design solutions for complex engineering problems by Identifying and survey the relevant literature.
100003/CS622T.2-PSO2	**MEDIUM**	Identify and survey the relevant literature for acquiring programming efficiency by designing algorithms and applying standard practices.
100003/CS622T.2-PSO3	**MEDIUM**	Identify and survey the relevant literature to apply the fundamentals of computer science in competitive research.
100003/CS622T.3-PO1	**HIGH**	Perform requirement analysis, identify design methodologies by using modern tools & advanced programming techniques and by applying the knowledge of mathematics, science, engineering fundamentals.
100003/CS622T.3-PO2	**HIGH**	Identify, formulate, review research literature for requirement analysis, identify design methodologies and develop adaptable & reusable solutions.

100003/CS622T.3-PO3	**HIGH**	Design solutions for complex engineering problems and perform requirement analysis, identify design methodologies.
100003/CS622T.3-PO4	**HIGH**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/CS622T.3-PO5	**HIGH**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools.
100003/CS622T.3-PO6	**MEDIUM**	Perform requirement analysis, identify design methodologies and assess societal, health, safety, legal, and cultural issues.
100003/CS622T.3-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts and Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions.
100003/CS622T.3-PO8	**HIGH**	Perform requirement analysis, identify design methodologies and develop adaptable & reusable solutions by applying ethical principles and commit to professional ethics.
100003/CS622T.3-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
100003/CS622T.3-PO10	**MEDIUM**	Communicate effectively with the engineering community and with society at large to perform requirement analysis, identify design methodologies.
100003/CS622T.3-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering requirement analysis by identifying design methodologies.
100003/CS622T.3-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by analysis, identify design methodologies and develop adaptable & reusable solutions.
100003/CS622T.3-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and prior to that perform requirement analysis, identify design methodologies.
100003/CS622T.4-PO1	**MEDIUM**	Prepare technical report and deliver presentation by applying the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.4-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by preparing technical report and deliver presentation.

100003/CS622T.4-PO3	**MEDIUM**	Prepare Design solutions for complex engineering problems and create technical report and deliver presentation.
100003/CS622T.4-PO4	**MEDIUM**	Use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions and prepare technical report and deliver presentation.
100003/CS622T.4-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and Prepare technical report and deliver presentation.
100003/CS622T.4-PO8	**HIGH**	Prepare technical report and deliver presentation by applying ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
100003/CS622T.4-PO9	**HIGH**	Prepare technical report and deliver presentation effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings.
100003/CS622T.4-PO10	**HIGH**	Communicate effectively with the engineering community and with society at large by prepare technical report and deliver presentation.
100003/CS622T.4-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work by prepare technical report and deliver presentation.
100003/CS622T.4-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change by prepare technical report and deliver presentation.
100003/CS622T.4-PSO1	**MEDIUM**	Prepare a technical report and deliver presentation to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas.
100003/CS622T.4-PSO2	**MEDIUM**	To acquire programming efficiency by designing algorithms and applying standard practices in software project development and to prepare technical report and deliver presentation.
100003/CS622T.4-PSO3	**MEDIUM**	To apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs by preparing technical report and deliver presentation.
100003/CS622T.5-PO1	**HIGH**	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
100003/CS622T.5-PO2	**HIGH**	Identify, formulate, review research literature, and analyze complex engineering problems by applying engineering and management principles to achieve the goal of the project.

100003/CS622T.5-PO3	**HIGH**	Apply engineering and management principles to achieve the goal of the project and to design solutions for complex engineering problems and design system components or processes that meet the specified needs.
100003/CS622T.5-PO4	**MEDIUM**	Apply engineering and management principles to achieve the goal of the project and use research-based knowledge including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
100003/CS622T.5-PO5	**MEDIUM**	Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO6	**MEDIUM**	Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues and the consequent responsibilities by applying engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO7	**MEDIUM**	Understand the impact of the professional engineering solutions in societal and environmental contexts, and apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO8	**HIGH**	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice and to use the engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO9	**MEDIUM**	Function effectively as an individual, and as a member or leader in teams, and in multidisciplinary settings and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO11	**MEDIUM**	Demonstrate knowledge and understanding of engineering and management principles and apply these to one's own work, as a member and leader in a team. Manage projects in multidisciplinary environments and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PO12	**HIGH**	Recognize the need for, and have the preparation and ability to engage in independent and lifelong learning in the broadest context of technological change and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PSO1	**MEDIUM**	The ability to identify, analyze and design solutions for complex engineering problems in multidisciplinary areas. Apply engineering and management principles to achieve the goal of the project.

100003/CS622T.5-PSO2	**MEDIUM**	The ability to acquire programming efficiency by designing algorithms and applying standard practices in software project development to deliver quality software products meeting the demands of the industry and to apply engineering and management principles to achieve the goal of the project.
100003/CS622T.5-PSO3	**MEDIUM**	The ability to apply the fundamentals of computer science in competitive research and to develop innovative products to meet the societal needs thereby evolving as an eminent researcher and entrepreneur and apply engineering and management principles to achieve the goal of the project.