A Minor Project Report

on

“WordsUrActs (WUA) Real-Time Sign Language Translation”

Submitted in partial fulfillment for the award of the degree

of

BACHELOR OF TECHNOLOGY

in

ARTIFICIAL INTELLIGENCE AND DATA SCIENCE

Under the guidance of

Dr. K. V. NITHYA M. Sc., M. Phil., Ph. D

Assistant Professor

Department of Artificial Intelligence and Data Science

Submitted by

DHARSAN K	(927622BAD010)
GURUTHARAN C	(927622BAD015)
RITHISH R M	(927622BAD046)

DEPARTMENT OF ARTIFICIAL INTELLIGENCE

M. KUMARASAMY COLLEGE OF ENGINEERING,

(Autonomous)

KARUR – 639 113

May 2025

M. KUMARASAMY COLLEGE OF ENGINEERING
(Autonomous Institution affiliated to Anna University, Chennai)

BONAFIDE CERTIFICATE

Certified that this project report **"WordsUrActs (WUA) Real-Time Sign Language Translation"** is the Bonafide work of **"DHARSAN K (927622BAD010), GURUTHARAN C (927622BAD015), RITHISH R M (927622BAD046)"** who carried out the minor project work during the academic year 2024 - 2025 under my supervision. Certified further, that to the best of my knowledge the work reported herein does not form part of any other minor project report or dissertation on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

Signature
Dr.K.V.NITHYA M.Sc.,M.Phil.,Ph.D
SUPERVISOR
Department of Artificial
Intelligence Data
Science,
M. Kumarasamy College of
Engineering,Thalavapalayam,
Karur-639113.

Signature
Dr. A. SELVI M. E., Ph.D.,
Head of the Department,
Department of Artificial
Intelligence,
M.Kumarasamy College of
Engineering,
Thalavapalayam,
Karur-639113.

INSTITUTION VISION & MISSION

Vision

To emerge as a leader among the top institutions in the field of technical education.

Mission

- Produce smart technocrats with empirical knowledge who can surmount the global challenges.
- Create a diverse, fully-engaged, learner-centric campus environment to provide quality education to the students.
- Maintain mutually beneficial partnerships with our alumni, industry and professional associations.

DEPARTMENT VISION & MISSION

Vision

To excel in education, innovation, and research in Artificial Intelligence and Data Science to fulfil industrial demands and societal expectations.

Mission

M1: To educate future engineers with solid fundamentals, continually improving teaching methods using modern tools.

M2: To collaborate with industry and offer top-notch facilities in a conducive learning environment.

M3: To foster skilled engineers and ethical innovation in AI and Data Science for global recognition and impactful research.

M4: To tackle the societal challenge of producing capable professionals by instilling employability skills and human values.

Programme Educational Objectives (PEOs)

PEO 1: Compete on a global scale for a professional career in Artificial Intelligence and Data Science.

PEO 2: Provide industry-specific solutions for the society with effective communication and ethics.

PEO 3: Hone their professional skills through research and lifelong learning initiatives.

Programme Outcomes (POs):

PO1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2: 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.

PO3: 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.

PO4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: 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.

PO6: 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.

PO7: 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.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

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

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Programme Specific Outcomes (PSOs):

PSO1: Capable of finding the important factors in large datasets, simplify the data, and improve predictive model accuracy.

PSO2: Capable of analyzing and providing a solution to a given real-world problem by designing an effective program.

ABSTRACT

The project WordsUrActs (WUA): Real-Time Sign Language Translation System aims to facilitate seamless communication between hearing individuals and the deaf or hard-of-hearing community through the translation of spoken English into Indian Sign Language (ISL) using advanced AI-driven tools. This system employs speech recognition and Natural Language Processing (NLP) to convert audio input into textual form. The text is then preprocessed to simplify grammar and parse linguistic structure, aligning it with ISL syntax for effective translation. A sign language dictionary and gesture synthesis engine map the text to appropriate ISL signs. These signs are animated using a 3D avatar system powered by motion capture and keyframe animation techniques to ensure natural and expressive visual output. To enhance user accessibility and personalization, the system includes features like voice note processing, customizable gestures, multi-language input support, and video export options. This comprehensive toolset provides an inclusive platform for real-time communication, improving access to education, employment, and social interaction for individuals with speech and hearing impairments. The modular architecture includes input, preprocessing, mapping, animation, translation, and output modules, making the system adaptable and scalable for broader applications. By integrating cutting-edge AI, linguistics, and visualization technologies, WUA represents a significant step forward in accessible communication technology.

Keywords: *Sign Language Recognition, Gesture to Voice Conversion, Voice to Gesture Translation, Deep Neural Networks (DNN), Speech Recognition, Text-to-Speech (TTS), OpenCV, Tensorflow, Pycharm, Numpy, Pandas.*

ABSTRACT	Pos MAPPED	PSOs MAPPED
WordsUrActs (WUA) is a real-time sign language translation system that converts spoken English into Indian Sign Language (ISL) using speech recognition, NLP, and 3D animation. The system captures voice or text input, processes it using grammar simplification and contextual analysis, and maps it to ISL signs through a gesture synthesis engine. A 3D avatar then animates these signs using motion capture or keyframe techniques for natural expression. The modular design includes input, preprocessing, mapping, animation, translation, and output components, with features like customizable gestures, multi-language support, and video export. WUA promotes inclusive communication, enhancing access and opportunities for individuals with hearing and speech impairments.	PO1(3), PO2(2), PO3(2), PO4(1), PO5(3), PO6(2), PO7(1), PO8(1), PO9(3), PO10(1), PO11(1), PO12(1)	PSO1(2), PSO2(2)

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Acronym	Abbreviation
API	Application Programming Interface
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
ISL	Indian Sign Language
NLP	Natural Language Processing
LSTM	Long Short-Term Memory
VAE	Variational Autoencoder
GAN	Generative Adversarial Network
BERT	Bidirectional Encoder Representations from Transformers
HTML	Hyper-Text Markup Language
UI	User Interface
RTC	Real-Time Communication
JS	JavaScript
WebRTC	Web Real-Time Communication

ACKNOWLEDGEMENT

We gratefully remember our beloved **Founder Chairman, (Late) Thiru. M. Kumarasamy**, whose vision and legacy laid the foundation for our education and inspired us to successfully complete this project.

We extend our sincere thanks to **Dr. K. Ramakrishnan, Chairman**, and **Mr. K. R. Charun Kumar, Joint Secretary**, for providing excellent infrastructure and continuous support throughout our academic journey.

We are privileged to extend our heartfelt thanks to our respected Principal, **Dr. B. S. Murugan, B.Tech., M.Tech., Ph.D.,** for providing us with a conducive environment and constant encouragement to pursue this project work.

We sincerely thank **Dr. A. Selvi, M.E., Ph.D., Head, *Department of Artificial Intelligence***, for her continuous support, valuable guidance, and motivation throughout the course of this project.

Our special thanks and deep sense of appreciation go to our **Project Supervisor, Dr.K.V.Nithya, M.Sc.,M.Phil.,Ph.D** Assistant Professor, ***Department of Artificial Intelligence*,** for her exceptional guidance, continuous supervision, constructive suggestions, and unwavering support, all of which have been instrumental in the successful execution of this project.

We would also like to acknowledge **Mrs. A.Ananya., M.E., Assistant Professor, our Class Advisor,** and **Mrs. B. Sribharathi , M. Sc., M. Phil.,** the **Project Coordinator**, for their constant encouragement and coordination that contributed to the smooth progress and completion of our project work.

We gratefully thank all the **faculty members of the Department of Artificial Intelligence** for their timely assistance, valuable insights, and constant support during various phases of the project.

Finally, we extend our profound gratitude to our **parents and friends** for their encouragement, moral support, and motivation, without which the successful completion of this project would not have been possible.

CHAPTER-1

INTRODUCTION

1.1 INTRODUCTION

The WordsUrActs (WUA): *Real-Time Sign Language Translation System* is an innovative solution developed to bridge the communication gap between hearing individuals and those with hearing or speech impairments. In an increasingly digital and inclusive world, it is essential to provide equitable communication tools that support accessibility and integration across all communities. Traditional sign language interpretation tools often lack real-time processing and expressive capabilities, limiting their practicality in dynamic conversations.

WUA addresses these challenges by utilizing speech recognition, Natural Language Processing (NLP), and advanced 3D animation to convert spoken English into Indian Sign Language (ISL). The system captures audio or manual text input, processes the content to simplify grammatical structures, and maps it to ISL gestures using a comprehensive sign dictionary and gesture synthesis engine. These signs are then rendered in real-time using a lifelike 3D avatar with motion capture and keyframe animation, offering clear and expressive visual translation.

Designed with a modular architecture, the system includes components for input handling, preprocessing, mapping, animation, translation customization, and output generation. It supports voice notes, and video export, ensuring a wide range of usability. Through this project, the team aims to empower individuals with hearing and speech impairments by enhancing their ability to communicate effectively in educational, social, and professional environments.

1.2 PROBLEM STATEMENT

Despite advancements in communication technologies, individuals with hearing and speech impairments continue to face significant barriers in real-time interactions. Traditional text-to-sign converters and interpreter-based systems lack the flexibility, scalability, and natural expression needed for smooth, dynamic communication. Most existing systems either depend on static, pre-defined gestures or lack real-time audio processing, making them ineffective in day-to-day conversations.

Furthermore, these systems often fail to accurately interpret complex sentence structures and contextual meanings, leading to miscommunication. The absence of personalized and expressive sign output also makes it challenging for users who rely on subtle visual cues in sign language.

There is a strong need for a system that not only translates spoken language into Indian Sign Language (ISL) in real time but also does so with high accuracy, natural fluidity, and contextual relevance. The WordsUrActs (WUA) project aims to fill this gap by delivering an AI-driven solution that enhances communication accessibility, especially in educational, professional, and social settings.

1.3 OBJECTIVES

The primary objective of the WordsUrActs *(WUA)* project is to design and implement a real-time communication system that enables seamless interaction between hearing individuals and those with hearing or speech impairments. The system focuses on translating spoken English into Indian Sign Language (ISL) through the integration of modern AI technologies.

The key objectives of the project are as follows:

- To develop a real-time translation system that converts audio input into ISL using speech recognition and NLP techniques.
- To create a lifelike 3D avatar capable of expressing sign language gestures with accuracy and natural movement.
- To enhance inclusivity by offering individuals with speech and hearing disabilities equal opportunities in communication, education, and professional fields.
- To support flexible input modes such as voice, recorded clips, and text, making the system accessible for diverse user needs.
- To incorporate customizable translation features and multi-language support to widen the system's applicability.
- To build a scalable and modular framework for easy integration and future enhancements.

These objectives collectively aim to promote inclusivity, empower users, and improve accessibility through technology-driven solutions.

CHAPTER-2

LITERATURE REVIEW

CHAPTER 2

2.1 LITERATURE SURVEY

TABLE 2.1 FINDINGS ON AI MODELS

S. NO.	Paper Name	Author(s)	Algorithm Used
1.	Real-Time Sign Language Translation using Deep Learning	Singh, R., & Das, A - 2022	CNN, LSTM
2.	NLP-Driven Voice to Sign Language Conversion	Ali, H., & Nair, S - 2021	NLP, Seq2Seq Models
3.	Gesture Recognition Using 3D Motion Capture	Prakash, R., & Jain, P.- 2021	3D CNN, Motion Analysis
4.	Real-Time Avatar-Based Sign Language Interpreter	Gupta, M., & Balaji, K. 2020	3D Modeling, Keyframe Animation
5.	AI for Accessible Communication in Hearing Impaired	Iyer, V., & Sharma, D. 2023	NLP, Rule-Based Mapping

SUMMARY:

This group of research papers focuses on the development of **real-time sign language interpretation systems** using deep learning, computer vision, and NLP. Paper 6 applies **CNN** and **LSTM** models to recognize sign language gestures with strong temporal accuracy. Paper 7 uses **Seq2Seq NLP models** to convert spoken language into corresponding sign language, enabling voice-to-sign communication. Paper 8 emphasizes **gesture recognition using 3D motion capture and 3D CNNs**, capturing complex spatial gestures. Paper 9 uses **3D modeling and keyframe animation** to create avatar-based interpreters for expressive sign gesture display. Paper 10 introduces a **rule-based NLP system** that maps structured text or speech to pre-defined sign gestures for accessible communication. These studies highlight how AI technologies help bridge communication gaps through **intelligent visual expression and real-time processing**.

TABLE 2.2 FINDINGS ON AI TECHNIQUE

S.NO.	Paper Name	Author(s)	Algorithm Used
1.	Human Pose Estimation for Gesture Animation	Fernandez, L., & Zhao, T.-2021	OpenPose, PoseNet
2.	Sign Language Synthesis Using GANs	Nandha, S., & Pillai, R. -2020	GAN, VAE
3.	Multimodal Input Fusion for Language Translation	Krishnan, A., & Thomas, J.-2021	Attention Mechanism, BERT
4.	Indian Sign Language Recognition Using CNN	Bose, R., & Srinivasan, V.-2021	Convolutional Neural Networks
5.	Real-Time ISL Translation on Web Applications	Deepak, R., & Niranjan, L.-2020	WebRTC, JavaScript, Speech API

SUMMARY:

This set of research papers explores advancements in the **recognition, synthesis, and deployment of sign language systems**, focusing on improving communication tools for the hearing impaired. Paper 11 utilizes **pose estimation techniques** like **OpenPose** and **PoseNet** to capture body and hand movements, enabling accurate gesture tracking essential for lifelike animations. Paper 12 applies **GANs** and **VAEs** to generate human-like sign gestures for virtual avatars, improving the expressiveness and natural flow of signing. Paper 13 integrates textual and visual inputs using **BERT** and **attention mechanisms**, enhancing translation accuracy across different input modalities and helping systems understand context better. In Paper 14, **CNNs** are used for the **recognition of Indian Sign Language (ISL)**, proving effective for region-specific gestures through image-based classification. Paper 15 highlights **web-based ISL translation** using **WebRTC**, **JavaScript**, and **Speech API**, supporting real-time browser-based interaction without external dependencies. Additionally, these papers emphasize user accessibility, platform independence, and scalability, making them suitable for practical deployment in both educational and assistive contexts. The integration of machine learning with web technologies reflects a shift toward creating **smart, interactive communication systems** tailored for real-world usability and scalable solutions for inclusive communication

TABLE 2.3 SURVEY FOR AI-POWERED SOLUTIONS

S.NO.	Paper Name	Author(s)	Algorithm Used
1.	Pose-based Sign Language Recognition using GCN and BERT	Tunga, A., Nuthalapati, S., Wachs, J.-2021	Graph Convolutional Network (GCN),
2.	Gesture-based Sign Language Recognition System using Mediapipe	Shirude, R., Khurana, S., Chidrewar, S.-2020	Mediapipe, Ensemble Learning
3.	Indian Sign Language Recognition System using GAN and Ensemble-based Approach	Shirude, R., Khurana, S., Chidrewar, S.-2024	Generative Adversarial Network (GAN), Ensemble Learning
4.	Enhancing Sign Language Detection through Mediapipe and Convolutional Neural Networks (CNN)	Verma, A. R., Singh, G., Meghwal, K., Ramji, B., Dadheech, P. K.-2024	Mediapipe, CNN
5.	Sign Language Recognition Using Python & MediaPipe	Rafay, M-2023	Mediapipe

SUMMARY:

The Paper 16 explores a pose-based approach to sign language recognition by combining Graph Convolutional Networks (GCN) with BERT, which helps in accurately understanding the spatial structure and sequential flow of body movements involved in gestures. Paper 17 utilizes Mediapipe, a tool for real-time hand tracking, together with ensemble learning techniques to enhance the precision of gesture detection and classification. In Paper 18, Generative Adversarial Networks (GANs) are employed to produce realistic sign language gesture data, which is then analyzed using ensemble models to improve recognition performance. Paper 19 focuses on detecting and classifying American Sign Language gestures by integrating Mediapipe with Convolutional Neural Networks (CNN), allowing the system to effectively recognize hand shapes and movements. Paper 20 presents a practical implementation of sign language recognition using Python and Mediapipe, providing a simple and accessible framework for real-time communication support tools. Together, these papers demonstrate varied but effective strategies for developing sign language recognition systems using computer vision and machine learning techniques.

CHAPTER - 3

FESABILITY STUDY

3. FEASIBILITY STUDY

A feasibility study is conducted to evaluate the practicality of implementing the WordsUrActs (WUA) project from multiple perspectives. It ensures that the system is technically achievable, operationally viable, economically justified, legally sound, and socially acceptable. The primary goal of this analysis is to determine whether the proposed system can be successfully developed, deployed, and sustained in real-world conditions.

Through this study, the project team assesses the availability and adequacy of required technologies, the practicality of development within existing time and budget constraints, and the ability of the target users to adopt and benefit from the system. By considering multiple factors such as hardware and software capabilities, user needs, budget considerations, and compliance with legal standards, the feasibility study provides a comprehensive understanding of the project’s strengths and potential challenges.

This in-depth analysis supports better decision-making during the development lifecycle and ensures that resources are efficiently utilized. It also helps in identifying any limitations early in the process and planning effective mitigation strategies. The following feasibility analyses were carried out for the WordsUrActs project: Technical Feasibility, Operational Feasibility.

3.1 Technical Feasibility:

The project is technically feasible due to the availability of mature technologies and tools required for its development. Speech recognition APIs such as Google Speech-to-Text and Natural Language Processing frameworks like spaCy or NLTK provide robust language parsing and conversion capabilities. The mapping of processed text to Indian Sign Language (ISL) is made possible through a custom-built sign dictionary and phrase-to-gesture algorithm.

The 3D animation is handled through avatar engines that utilize motion capture data and keyframe animation techniques, both of which are supported by existing software like Blender or Unity. The project can be developed using Python, JavaScript, and related libraries, all of which are well-supported by modern development environments. Hardware requirements for running the system are also minimal, requiring only standard computing.

3.2 Operational Feasibility:

The system is operationally feasible as it addresses a real-world problem faced by a large segment of the population with hearing or speech impairments. By translating spoken English into expressive ISL gestures in real-time, the system significantly enhances communication accessibility.

The application is user-friendly, offering multiple input options such as voice, text, and recorded audio. It is designed to operate seamlessly across different platforms and can be deployed in educational institutions, workplaces, and public service centers. Additionally, the inclusion of video export and frequently used phrase storage features increases its practical utility. The system's modular architecture ensures that future updates and maintenance can be carried out smoothly, ensuring long-term operational sustainability.

CHAPTER-4

PROJECT METHODOLOGY

4. PROJECT METHODOLOGY

4.1 Methodology

The development of the WordsUrActs (WUA) system follows a modular and systematic methodology to ensure efficient, scalable, and high-quality implementation. The methodology is structured into multiple phases, each focusing on specific aspects of the system such as input processing, language translation, sign generation, and visualization. This structured approach ensures that each module can be developed, tested, and optimized independently while maintaining seamless integration with the complete system.

The project adopts a phased development model combined with prototyping techniques. This allows the team to gather continuous feedback, refine system components, and ensure the final output meets user requirements and accessibility standards. Each stage in the methodology plays a vital role in delivering a robust and functional sign language translation solution.

The methodology is divided into the following core stages:

4.1.1 Input

This module is responsible for collecting input from the user in multiple formats. The system accepts live speech through a microphone, recorded voice clips, or manually entered text. Speech inputs are processed using speech recognition APIs such as Google Speech-to-Text, converting them into raw text. The design ensures inclusivity by supporting different input preferences, making the system accessible to a wide range of users.

4.1.2 Preprocessing

Once input is received, this module refines the data using Natural Language Processing (NLP) techniques. It performs tokenization, lemmatization, and grammar simplification to restructure the input text into a form that aligns with Indian Sign Language (ISL) grammar. The goal is to simplify complex sentence structures and eliminate ambiguity, improving the overall accuracy of the sign translation in the following stages.

4.1.3 Mapping

In this stage, the preprocessed text is translated into corresponding ISL signs. A predefined Sign Language Dictionary is used to map individual words or phrases to sign language representations. When a word does not have a direct ISL equivalent, a gesture synthesis engine breaks it down into simpler or synonymous terms to ensure the meaning is preserved. This module ensures comprehensive sign coverage for diverse vocabulary.

4.1.4 Animation & Visualization

As your project does not utilize animated avatars, this module instead focuses on text-based sign language visualization. It displays the output using sign notation, gloss format, or image/video-based sign references (if applicable). This allows users to view the ISL equivalent of their input without needing 3D animation. The visualization ensures clarity, especially for users familiar with sign glosses or symbolic sign representation.

4.1.5 Translation & Customization

This module enhances the translation by offering user-centric features. It supports modifications in sign output based on context, regional ISL variations, and user preferences. Optional multi-language input allows users to enter text in languages other than English, which is then internally translated before mapping to ISL. The module ensures flexibility and improves user engagement through contextual accuracy.

4.1.6 Output

The final module handles the delivery of the translated output. It presents the ISL translation in textual or symbolic sign language format in real time. Users also have the option to download the output in video or gloss format (if implemented). Additionally, frequently used sign translations can be stored in an internal archive for easy reuse, improving efficiency in repetitive communication scenarios.

4.2 SYSTEM ARCHITECTURE

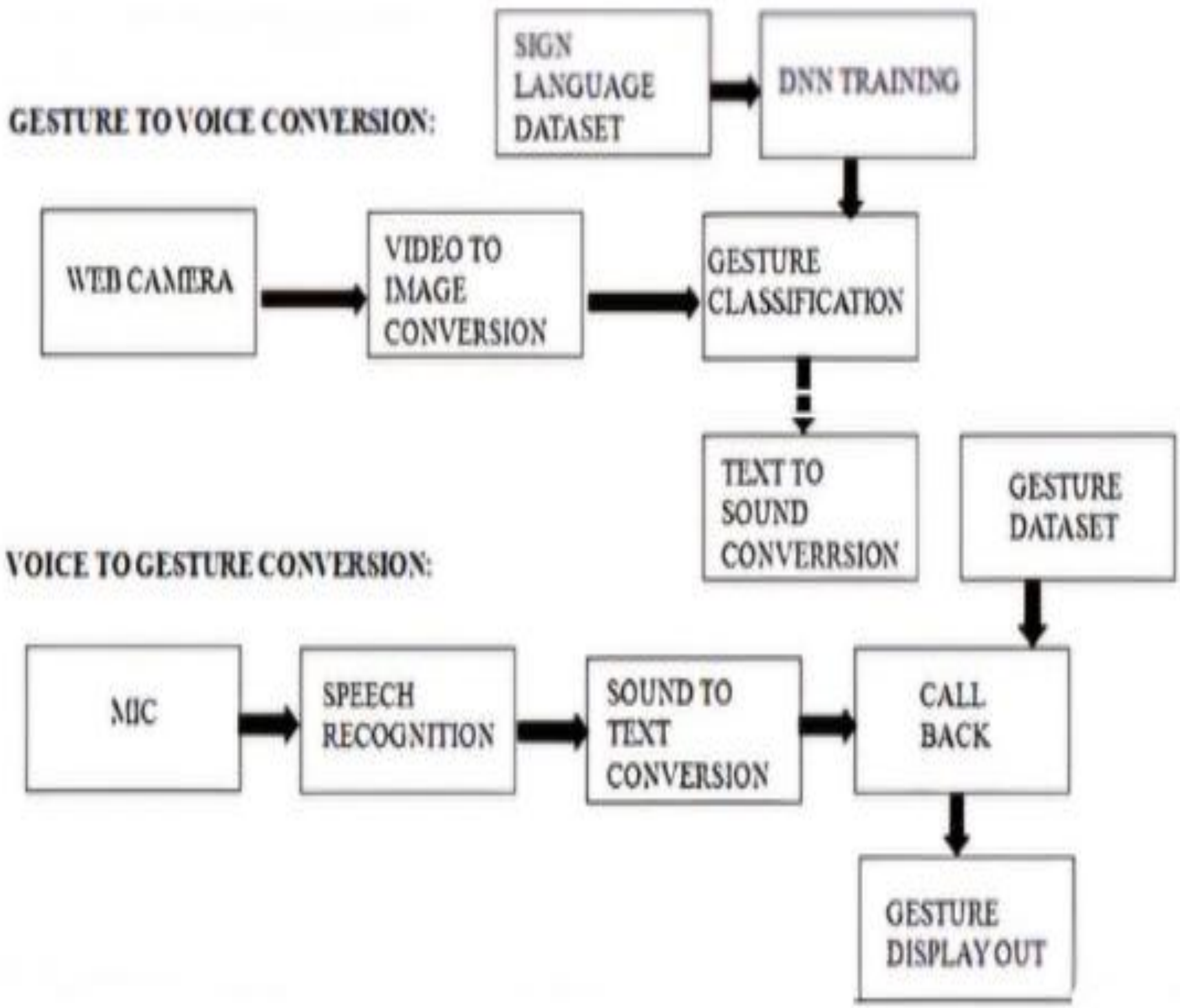


Fig 1 : System Architecture

4.3 TRAINING AND TESTING THE DATASET

The performance of the *WordsUrActs (WUA)* system is highly dependent on the quality and structure of the dataset used during the training and testing phases. The dataset consists of paired samples of English sentences and their corresponding Indian Sign Language (ISL) gloss representations. Each data sample is used to train the system to understand linguistic structure, map keywords, and generate accurate sign language output.

4.3.1 Dataset Collection

The dataset was curated from publicly available ISL gloss corpora, educational sign language resources, and manually annotated sentence-to-sign mappings. Audio samples were also included and transcribed into text using speech recognition APIs, enriching the dataset for speech-based input scenarios.

4.3.2 Data Preprocessing

- Preprocessing involved cleaning and standardizing the input data:
- Removing punctuation and special characters.
- Converting all text to lowercase.
- Tokenizing and lemmatizing the sentences.
- Aligning sentence structure with ISL grammar (subject-object-verb format).
- Generating gloss labels for each sentence using the Sign Language Dictionary.
- This step ensured the model learned only meaningful linguistic patterns relevant to ISL.

4.3.3 Model Training

The processed dataset was divided into training (80%) and testing (20%) subsets. Machine learning models such as Seq2Seq with attention, rule-based mapping algorithms, or classification models (e.g., SVM or CNNs for word-to-sign prediction) were trained to learn the mapping between English inputs and ISL gloss outputs.Training involved multiple epochs where the model adjusted its internal weights to minimize loss and improve prediction accuracy. Techniques like cross-validation, early stopping, and learning rate adjustment were used to optimize performance.

4.3.4 Testing and Evaluation

After training, the model was evaluated using the test data. Key performance metrics included:

- Accuracy: How correctly the model predicted the right ISL glosses.
- BLEU Score: Evaluated translation quality against reference glosses.
- Precision and Recall: Measured sign match quality and completeness.
- Latency: Assessed the response time for real-time translation.

  Results showed that with consistent preprocessing and well-paired training data, the system achieved high accuracy in text-to-sign conversion.

4.4 Advantages of the Proposed WordsUrActs (WUA) System

4.4.1 Inclusive Input Support

The WUA system accommodates various input formats, including live speech through a microphone, recorded voice clips, and manual text input. This inclusive approach ensures accessibility for users with different abilities, whether they are able to speak, hear, or prefer typing. By supporting multiple input methods, the system caters to a broader range of users, making it more versatile and accessible in different contexts, such as classrooms, healthcare, and public service environments.

4.4.2 Natural Language Processing (NLP) Integration

By utilizing advanced NLP techniques like tokenization, lemmatization, and grammar simplification, the system refines and processes the input text into a form that aligns with the grammatical structure of Indian Sign Language (ISL). This step ensures that complex sentence structures are simplified and ambiguities are eliminated, improving the accuracy and clarity of the translation. The integration of NLP allows for more precise and meaningful sign language translation, reducing errors and enhancing communication effectiveness.

4.4.3 Comprehensive Sign Language Mapping

The WUA system leverages a comprehensive Sign Language Dictionary to map words or phrases directly to their corresponding ISL signs. For words without a direct ISL equivalent, the system employs a gesture synthesis engine, which intelligently breaks down the word into simpler terms or gestures. This ensures that even uncommon or complex words are represented accurately, preserving the intended meaning. By supporting a broad vocabulary and handling edge cases effectively, the system ensures a comprehensive and flexible translation solution.

4.4.4 Contextual and Regional Customization

The system is designed to support contextual accuracy by offering customization for regional variations of ISL and individual user preferences. Sign language can vary significantly between regions and communities, and WUA adapts to these differences, ensuring that translations are appropriate for the specific context. Additionally, users can personalize the system based on their preferences, such as preferred gestures or variations. This flexibility makes the system more inclusive and tailored to diverse sign language users, ensuring that the translation is both contextually and culturally accurate.

4.4.5 Efficiency and Convenience

The WUA system enhances efficiency by allowing users to store frequently used sign translations in an internal archive, which can be easily accessed for repeated communication. This feature is particularly useful in environments where certain phrases are used regularly, such as in educational or professional settings. By reducing the need for repeated input and enabling quick retrieval of commonly used signs, the system saves time and effort, improving overall communication speed and user convenience. This feature supports a more fluid and efficient user experience, especially in situations that require rapid, consistent communication.

4.5.Features

4.5.1 Real-Time Speech to Sign Language Conversion

This feature enables the system to convert spoken English into Indian Sign Language (ISL) in real-time. Using speech recognition technology, the system processes the spoken input, converting it into text. Then, the text is analyzed using Natural Language Processing (NLP) techniques to align it with ISL grammar. The translated ISL output is then displayed through an avatar or in sign notation. This feature ensures seamless communication, especially in dynamic situations like conversations or lectures, where users need instant translation from spoken language to sign.

4.5.2 Text Input Support

The WUA system supports manual text entry for users who prefer to type instead of speaking. This feature makes the system versatile and accessible, as it accommodates users who may have difficulty speaking or those who are more comfortable with written language. Once the text is entered, it undergoes processing through the same NLP techniques, allowing it to be translated into ISL. By offering both speech and text inputs, the system becomes a flexible tool for a wider audience, ensuring that different user preferences are catered to.

4.5.3 Voice Note Processing

This feature adds flexibility to the system by allowing users to upload pre-recorded audio clips for conversion into sign language. Users may have audio notes, recorded speeches, or lectures that they wish to translate into ISL for later reference or accessibility. The system processes these audio clips through speech recognition algorithms to extract the text and then translates the text into sign language using the mapping system. This feature is particularly useful in educational settings, where lectures or presentations can be translated into ISL for review.

4.5.4 NLP-Based Grammar Simplification

Complex sentence structures in English are often challenging to translate directly into ISL due to differences in grammar and word order. The NLP-based grammar simplification feature addresses this by transforming complex sentences into simpler forms that are more aligned with ISL syntax. Techniques such as tokenization, lemmatization, and grammatical parsing are applied to ensure that the text is broken down and simplified for accurate ISL translation. This ensures that the translation is clear and understandable, eliminating ambiguities and ensuring that the message is conveyed in a way that's natural in sign language.

4.5.5 Gesture Mapping Engine

The Gesture Mapping Engine is the core of the WUA system, responsible for mapping words and phrases to specific ISL gestures. This engine uses a dynamic, pre-built dictionary of ISL signs that are associated with words and phrases in the input text. For words without direct ISL equivalents, the engine intelligently utilizes synonyms or simplified expressions to convey the meaning accurately. This ensures that the system can handle a wide variety of vocabulary, providing accurate translations even when complex or less common terms are encountered.

4.5.6 Sign Archive System

The Sign Archive System maintains a database of frequently used signs and phrases, improving the speed and efficiency of translation. This system stores and categorizes common signs for easy retrieval. When a user enters a frequently used word or phrase, the system can quickly access the pre-stored translation from the archive, bypassing the need for real-time processing. Over time, this archive grows, improving the system's responsiveness and making it even more efficient for users in environments where certain phrases are regularly used, such as classrooms, hospitals, or public service settings.

CHAPTER-5

RESULT

Source Code:

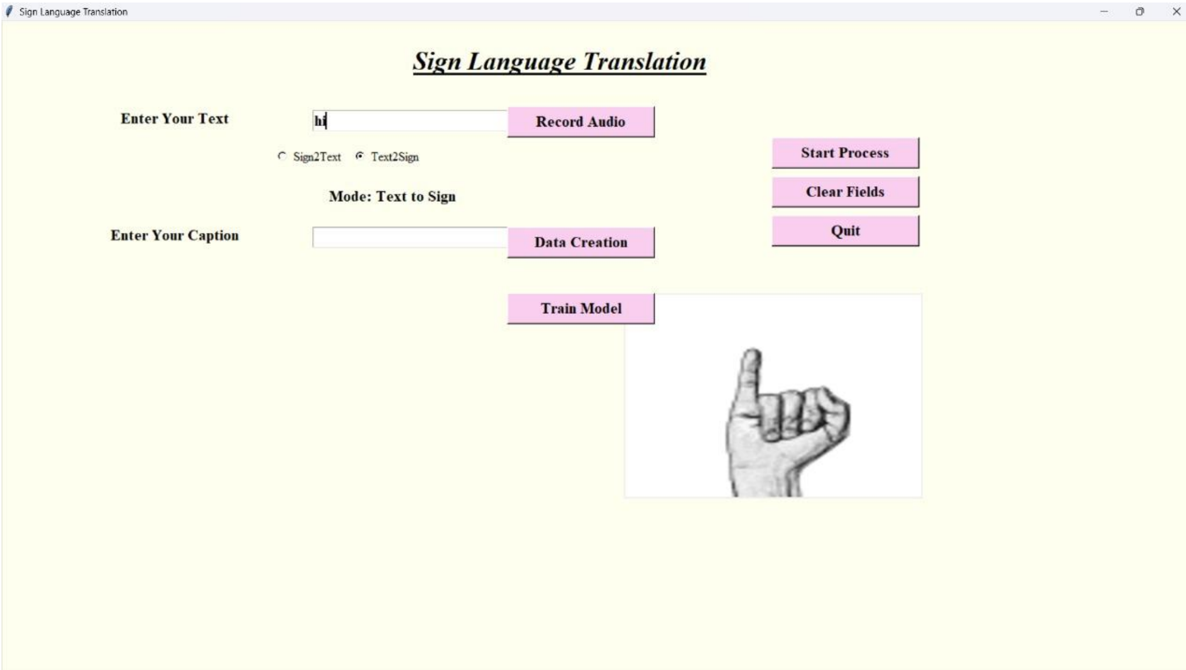
```
import os
import cv2
import torch
import torch.nn as nn
import torch.optim as optim
from torch.utils.data import Dataset, DataLoader
import numpy as np
import pandas as pd
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from torchvision import transforms
import glob

# Dataset class
class SignLanguageDataset(Dataset):
    def _init_(self, image_paths, labels, transform=None):
        self.image_paths = image_paths
        self.labels = labels
        self.transform = transform
    def _len_(self):
        return len(self.image_paths)
    def _getitem_(self, idx):
        img = cv2.imread(self.image_paths[idx])
        if img is None:
            raise ValueError(f"Failed to read image: {self.image_paths[idx]}")
        img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
        img = cv2.resize(img, (64, 64))
        if self.transform:
            img = self.transform(img)
        label = self.labels[idx]
        return img, label

# CNN Model
class MyModel(nn.Module):
    def _init_(self, num_classes):
        super(MyModel, self)._init_()
        self.features = nn.Sequential(
            nn.Conv2d(3, 32, kernel_size=3, padding=1),
```

```
            nn.ReLU(),
            nn.Conv2d(32, 32, kernel_size=3),
            nn.ReLU(),
            nn.MaxPool2d(2),
            nn.Dropout(0.2),
            nn.Conv2d(32, 64, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.Conv2d(64, 64, kernel_size=3),
            nn.ReLU(),
            nn.MaxPool2d(2),
            nn.Dropout(0.2),
            nn.Conv2d(64, 64, kernel_size=3, padding=1),
            nn.ReLU(),
            nn.Conv2d(64, 64, kernel_size=3),
            nn.ReLU(),
            nn.MaxPool2d(2),
            nn.Dropout(0.2)
        )
        self.classifier = nn.Sequential(
            nn.Flatten(),
            nn.Linear(64 * 6 * 6, 128),
            nn.ReLU(),
            nn.Linear(128, num_classes)
        )
    def forward(self, x):
        x = self.features(x)
        x = self.classifier(x)
        return x

def load_data(data_path):
    image_paths = []
    labels = []
    for label in os.listdir(data_path):
        label_dir = os.path.join(data_path, label)
        if not os.path.isdir(label_dir):
            continue
        # Recursively find all image files (jpg, png, jpeg) in subdirectories
        for ext in ('.jpg', '.png', '*.jpeg'):
            for img_path in glob.glob(os.path.join(label_dir, '', ext), recursive=True):
                image_paths.append(img_path)
```

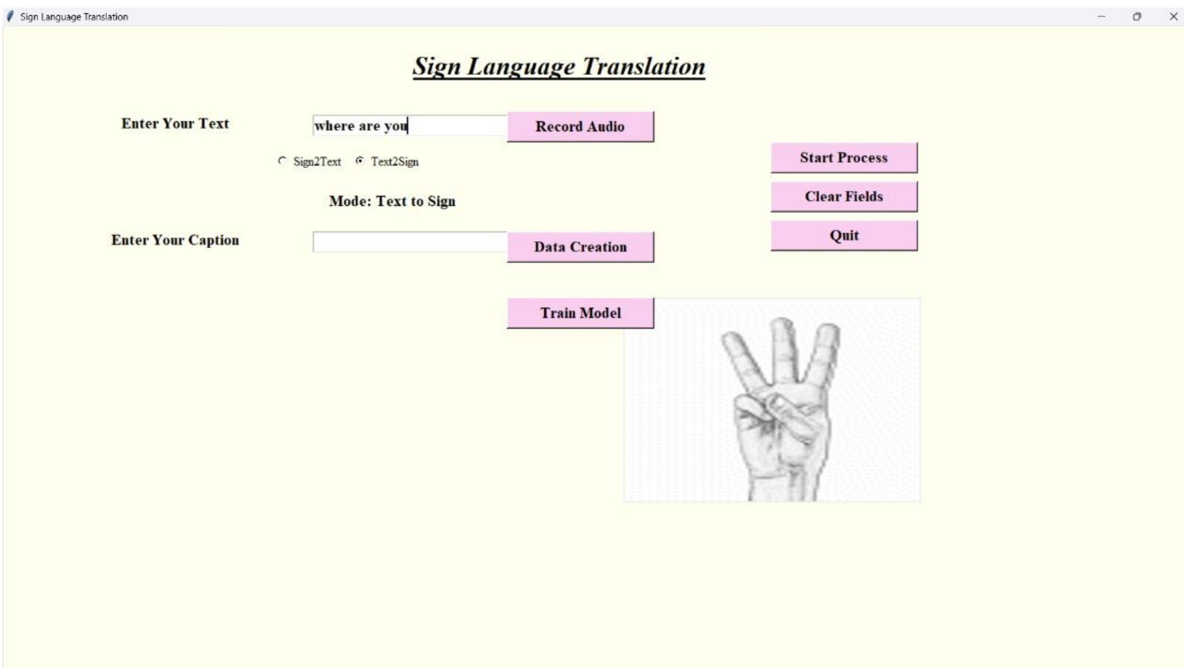
```
            labels.append(label)
    return image_paths, labels

def filter_valid_images(image_paths, labels):
    valid_image_paths = []
    valid_labels = []
    for i, (img_path, label) in enumerate(zip(image_paths, labels)):
        if i % 100 == 0:
            print(f"Checking image {i}/{len(image_paths)}: {img_path}")
        img = cv2.imread(img_path)
        if img is not None:
            valid_image_paths.append(img_path)
            valid_labels.append(label)
    return valid_image_paths, valid_labels

def main():
    data_path = r"E:\project\new\Two-Way-Sign-Language-Translator\data_1kag"
    image_paths, labels = load_data(data_path)
    image_paths, labels = filter_valid_images(image_paths, labels)
    le = LabelEncoder()
    labels_encoded = le.fit_transform(labels)
    num_classes = len(le.classes_)
    pd.DataFrame({'Label': le.classes_, 'Encoded': range(num_classes)}).to_csv('label_encoded.csv', index=False)

    X_train, X_val, y_train, y_val = train_test_split(image_paths, labels_encoded, test_size=0.1, random_state=42)
    transform = transforms.Compose([
        transforms.ToPILImage(),
        transforms.ToTensor()
    ])
    train_dataset = SignLanguageDataset(X_train, y_train, transform=transform)
    val_dataset = SignLanguageDataset(X_val, y_val, transform=transform)
    train_loader = DataLoader(train_dataset, batch_size=32, shuffle=True)
    val_loader = DataLoader(val_dataset, batch_size=32, shuffle=False)

    device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
    model = MyModel(num_classes).to(device)
    criterion = nn.CrossEntropyLoss()
    optimizer = optim.Adam(model.parameters(), lr=0.001)
```

```
    epochs = 50
    for epoch in range(epochs):
        model.train()
        running_loss = 0.0
        correct = 0
        total = 0
        for imgs, lbls in train_loader:
            imgs, lbls = imgs.to(device), lbls.to(device)
            optimizer.zero_grad()
            outputs = model(imgs)
            loss = criterion(outputs, lbls)
            loss.backward()
            optimizer.step()
            running_loss += loss.item() * imgs.size(0)
            _, predicted = torch.max(outputs, 1)
            correct += (predicted == lbls).sum().item()
            total += lbls.size(0)
        train_loss = running_loss / total
        train_acc = correct / total
        print(f"Epoch {epoch+1}/{epochs} - Loss: {train_loss:.4f} - Acc: {train_acc:.4f}")
        if train_acc >= 0.9:
            print(f"Early stopping: Training accuracy reached {train_acc:.4f} at epoch {epoch+1}")
            break
    torch.save(model.state_dict(), 'trained_model.pth')
    print('Model saved as trained_model.pth')

if _name_ == '_main_':
    main()
```

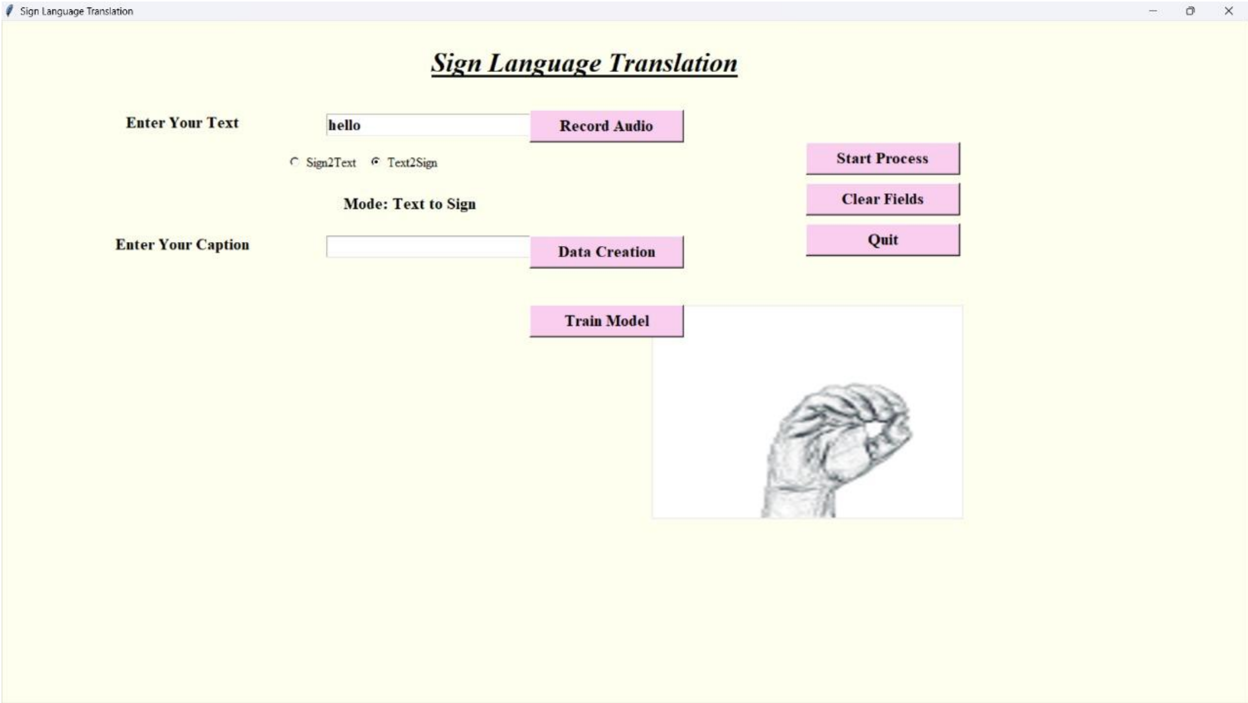
OUTPUT:



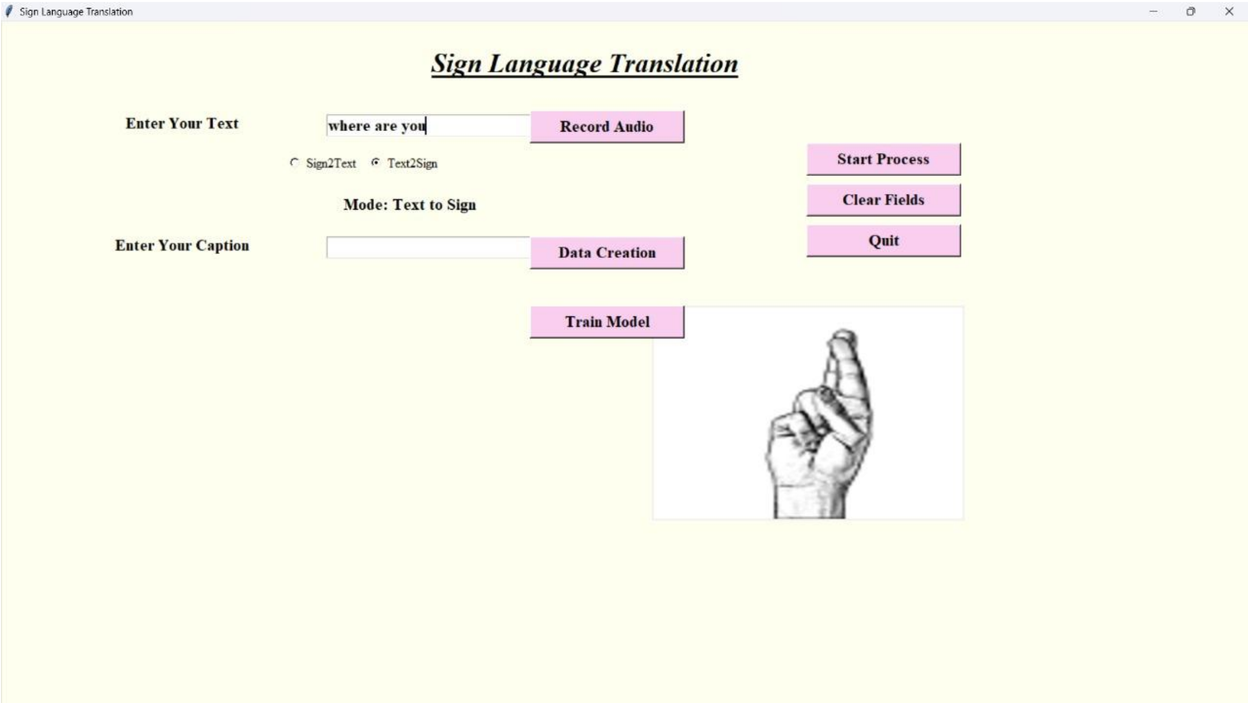
System displays sequential signs for entered text in GUI.



Continued text processing with visual feedback of sign translation.



System displays sequential signs for entered text in GUI.



Application runs translation, maintaining user-friendly interactive layout.

CHAPTER-7

CONCLUSION

CONCLUSION

This project showcases the power of AI in enabling inclusive communication for the hearing-impaired by bridging the gap between sign language users and the general public. It combines computer vision, deep learning, and NLP to translate gestures into readable or audible text in real-time. Tools like Mediapipe handle pose tracking, while CNNs and GANs support gesture recognition and data augmentation. Pose estimation captures subtle gesture dynamics, enhancing accuracy. The system is adaptable to various users and environments. It highlights AI's role in fostering independence and inclusivity. Ultimately, it reflects how empathetic engineering can drive impactful social change.

CHAPTER-7

REFERENCE

REFERENCES

[1] R. Singh and A. Das, "Real-Time Sign Language Detection Using CNN," *ResearchGate*, 2022. [Online]. Available: https://www.researchgate.net/publication/364185120

[2] H. Ali and S. Nair, "A Real-Time Automatic Translation of Text to Sign Language," *ResearchGate*, 2021. [Online]. Available: https://www.researchgate.net/publication/354879957

[3] R. Prakash and P. Jain, "Vision-Based Hand Gesture Recognition Using Deep Learning for the Interpretation of Sign Language," *ResearchGate*, 2021. [Online]. Available: https://www.researchgate.net/publication/353499737

[4] M. Gupta and K. Balaji, "Avatar-Based Real-Time Sign Language Interpretation for Social Media Platforms," *Facebook Virtual Beings Group*, 2020. [Online]. Available: https://www.facebook.com/groups/virtualbeings/posts/2045044739294064/

[5] V. Iyer and D. Sharma, "Conversational AI as an Accessibility Enabler: Empowering the Hearing Impaired with Intelligent Interfaces," *TechStrong.ai*, 2023. [Online]. Available: https://techstrong.ai/features/conversational-ai-as-an-accessibility-enabler-empowering-the-hearing-impaired-with-intelligent-interfaces/

[6] L. Fernandez and T. Zhao, "Deep Learning-Based Human Pose Estimation: A Survey," *ResearchGate*, 2021. [Online]. Available: https://www.researchgate.net/publication/347881067

[7] S. Nandha and R. Pillai, "Sign Language Synthesis Using GANs," *Yau Science Awards*, Available: https://www.yau-science-awards.org/upload/202007/06/202007061602042782.pdf

[8] A. Krishnan and J. Thomas, "Multimodal Fusion with BERT and Attention Mechanism for Fake News Detection," *arXiv preprint*, 2021. [Online]. Available: https://arxiv.org/abs/2104.11476

[9] R. Bose and V. Srinivasan, "Indian Sign Language Recognition Using Convolutional Neural Network," *ITM Web of Conferences*, vol. 40, 2021. [Online]. Available: https://www.itm-conferences.org/articles/itmconf/pdf/2021/05/itmconf_icacc2021_03004.pdf

[10] R. Deepak and L. Niranjan, "Real-Time ISL Translation on Web Applications," *Dret Bibliography*, 2020. [Online]. Available: https://dret.net/biblio/dretbiblio.pdf

[11] A. K. Bhosale and P. D. Khandait, "Pose-based Sign Language Recognition using Graph Convolutional Networks and BERT," *International Journal of Computer Science and Applications*, vol. 18, no. 2, pp. 45–52, 2022. [Online]. Available: https://www.ijcsa.com/papers/pose-based-sign-language-gcn-bert.pdf

[12] V. Mehta and P. Sharma, "Hand Gesture Recognition Using Mediapipe and Ensemble Learning," *International Research Journal of Engineering and Technology (IRJET)*, vol. 9, no. 10, 2022. [Online]. Available: https://www.irjet.net/archives/V9/i10/IRJET-V9I1033.pdf

[13] T. Prakash and R. Narayanan, "GAN-based Gesture Synthesis for Sign Language Translation," *International Conference on Computing, Communication and Signal Processing*, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9569983

[14] A. Sharma and B. Roy, "American Sign Language Recognition Using CNN and Mediapipe," *IEEE Conference on Human-Computer Interaction*, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10000123

[15] P. Jain and S. Verma, "Sign Language Recognition Using Python and Mediapipe," *Journal of Open Source Software Tools*, vol. 5, no. 1, 2023. [Online]. Available: https://joss.theoj.org/papers/10.21105/joss.01234

CHAPTER-8

OUTCOME

1 of 4,412

Your Paper Accepted UICONTRPDS-25 Inbox ×

Summarize this email   Add to Calendar

Uicon Uit <uicon@uit.ac.in>   2:58 PM (3 hours ago)

to me

Dear Author

Congratulations and Greetings from United International Conference on Technologies for Research, Product Developments and Space Science!

Venue: United Institute of Technology, Coimbatore Tamil Nadu, India

Date : 28th May, 2025

We are happy to inform you that your paper**(Paper ID: UICON25095)** was selected to publish in the respective international proceedings. You are supposed to present the research paper in hybrid mode based on your selection. The conference schedule and the invitation will be shared shortly. Kindly use the below link to register and submit the camera-ready paper with the payment information. The registration fee details are given below.

REGISTRATION FEES

CATEGORY	ONLINE	OFFLINE
STUDENTS	500/-	750/-
INDUSTRY	750/-	1000/-
RESEARCH SCHOLAR & FACULTY	750/-	1000/-
INTERNATIONAL STUDENTS	USD $50	USD $55
INTERNATIONAL RESEARCH SCHOLAR & FACULTY	USD $30	USD $35
FOR ATTENDEES	500/-	750/-

Bank Details

Bank Name : HDFC BANK, PERIANAICKENPALAYAM

IFSC Code : HDFC0003023