

**A REPORT
ON
Indian Sign Language to Text/Speech translation**

Submitted by,

**Mr. Hasan Raza B A - 20211CAI0092
Mr. Chakradhar Reddy - 20211CAI0156
Mr. Naheel N Akhtar - 20211CAI0142
Mr. Nida Aiyman - 20211CAI0085**

Under the guidance of,

Mr. Likhith S R

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At



GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY

BENGALURU

MAY 2025

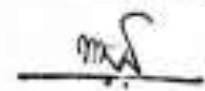
PRESIDENCY UNIVERSITY

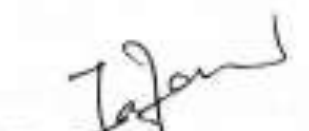
**PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND
ENGINEERING**


CERTIFICATE

This is to certify that the Internship/Project report “Indian Sign Language to Text/Speech translation” being submitted by “Hasan Raza B A , Chakradhar Reddy, Naheel N Akhtar, Nida Aiyman” bearing roll number “20211CAI0092, 20211CAI00156, 20211CAI0142, 20211CAI0085” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.


Mr. Likhith S R
Assistant Professor
PSCS
Presidency University


Dr. MYDHILI NAIR
Associate Dean
PSCS
Presidency University


Dr. Zafar Ali Khan
Professor & HoD
PSCS
Presidency University


Dr. SAMEERUDDIN KHAN
Pro-Vice Chancellor - Engineering
Dean -PSCS / PSIS
Presidency University

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I hereby declare that the work, which is being presented in the report entitled "**Indian Sign Language to Text/Speech translation**" in partial fulfillment for the award of Degree of **Bachelor of Technology in Computer Science and Engineering**, is a record of my own investigations carried under the guidance of **Mr. Likhith S R, Assistant Professor, Presidency School of Computer Science and Engineering, Presidency University, Bengaluru.**

I have not submitted the matter presented in this report anywhere for the award of any other Degree.


16/5/25




Hasan Raza B A (20211CAI0092)
Chakradhar Reddy(20211CAI00156)
Naheel N Akhtar(20211CAI0142)
Nida Aiyman(20211CAI0085)

ABSTRACT

Sign language is a crucial mode of communication for the deaf and hard-of-hearing community, employing hand gestures, facial expressions, and body movements to convey meaning. Indian Sign Language (ISL) is the primary sign language used in India; however, a significant communication gap exists between ISL users and those who rely on spoken or written language. To address this barrier, technology-driven solutions such as ISL-to-text/speech translation systems are being developed. These systems leverage artificial intelligence (AI), computer vision, and natural language processing (NLP) to recognize and interpret hand gestures, converting them into meaningful text or speech. Recent advancements in deep learning, gesture recognition, and wearable sensor technologies have significantly enhanced the accuracy and efficiency of ISL translation systems. These innovations not only promote accessibility and inclusivity but also empower individuals with hearing impairments by providing real-time translation tools for various domains, including education, employment, healthcare, and social interactions. By bridging the communication divide, ISL translation technologies play a vital role in fostering equal opportunities and seamless integration of the deaf and hard-of-hearing community into mainstream society.

**A REPORT
ON
Indian Sign Language to Text/Speech translation**

Submitted by,

**Mr. Hasan Raza B A - 20211CAI0092
Mr. Chakradhar Reddy - 20211CAI0156
Mr. Naheel N Akhtar - 20211CAI0142
Mr. Nida Aiyman - 20211CAI0085**

Under the guidance of,

Mr. Likhith S R

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING

At



PRESIDENCY UNIVERSITY

BENGALURU

MAY 2025

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the Internship/Project report “**Indian Sign Language to Text/Speech translation**” being submitted by “Hasan Raza B A , Chakradhar Reddy, Naheel N Akhtar, Nida Aiyman” bearing roll number “20211CAI0092, 20211CAI00156, 20211CAI0142, 20211CAI0085” in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bonafide work carried out under my supervision.

Mr. Likhith S R
Assistant Professor
PSIS
Presidency University

Dr. Zafar Ali Khan
Professor & HoD
PSCS
Presidency University

Dr. MYDHILI NAIR
Associate Dean
PSCS
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-Vice Chancellor - Engineering
Dean –PSCS / PSIS
Presidency University

PRESIDENCY UNIVERSITY

PRESIDENCY SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

I hereby declare that the work, which is being presented in the report entitled “**Indian Sign Language to Text/Speech translation**” in partial fulfillment for the award of Degree of **Bachelor of Technology in Computer Science and Engineering**, is a record of my own investigations carried under the guidance of **Mr. Likhith S R, Assistant Professor, Presidency School of Computer Science and Engineering, Presidency University, Bengaluru.**

I have not submitted the matter presented in this report anywhere for the award of any other Degree.

Hasan Raza B A (20211CAI0092)

Chakradhar Reddy(20211CAI00156)

Naheel N Akhtar(20211CAI0142)

Nida Aiyman(20211CAI0085)

ABSTRACT

Sign language is a crucial mode of communication for the deaf and hard-of-hearing community, employing hand gestures, facial expressions, and body movements to convey meaning. Indian Sign Language (ISL) is the primary sign language used in India; however, a significant communication gap exists between ISL users and those who rely on spoken or written language. To address this barrier, technology-driven solutions such as ISL-to-text/speech translation systems are being developed. These systems leverage artificial intelligence (AI), computer vision, and natural language processing (NLP) to recognize and interpret hand gestures, converting them into meaningful text or speech.

Recent advancements in deep learning, gesture recognition, and wearable sensor technologies have significantly enhanced the accuracy and efficiency of ISL translation systems. These innovations not only promote accessibility and inclusivity but also empower individuals with hearing impairments by providing real-time translation tools for various domains, including education, employment, healthcare, and social interactions. By bridging the communication divide, ISL translation technologies play a vital role in fostering equal opportunities and seamless integration of the deaf and hard-of-hearing community into mainstream society.

ACKNOWLEDGEMENTS

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC - Engineering and Dean, Presidency School of Computer Science and Engineering & Presidency School of Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Dean **Dr. Mydhili Nair**, Presidency School of Computer Science and Engineering, Presidency University, and **Dr. Zafar Ali Khan**, Head of the Department, Presidency School of Computer Science and Engineering, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Mr. Likhith S R, Assistant Professor** and Reviewer **Mr. Likhith S R, Assistant Professor**, Presidency School of Computer Science and Engineering, Presidency University for his inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the internship work.

We would like to convey our gratitude and heartfelt thanks to the CSE7301 Internship/University Project Coordinator **Mr. Md Ziaur Rahman and Dr. Sampath A K**, department Project Coordinators **Dr. Afroz Pasha** and Git hub coordinator **Mr. Muthuraj**.

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

Hasan Raza B A(1)
Chakradhar Reddy(2)
Naheel N Akhtar(3)
Nida Aiyman(4)

LIST OF TABLES

Sl. No.	Table Name	Table Caption	Page No.
1	Table 2.1	Study of Tools and Methodologies	3

LIST OF FIGURES

Sl. No.	Figure Name	Caption	Page No.
1	Fig 4.1	ISL signs captured through a webcam	16
2	Fig.4.2.	Hand Landmarks on a captured image	18
3	Fig 6. 1	System Design Diagram	22
4	Fig. 7.1.	Time line of the project	24
5	Fig. 9.1.	Performance of the model	28
6	Fig 11. 1	Input from detected signs	41
7	Fig 11. 2	Input translated to Kannada	41
8	Fig 11. 3	Input translated into Hindi	42
9	Fig 11. 4	Input translated into French	42

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	ACKNOWLEDGEMENTS	vi
1.	INTRODUCTION	1
2.	LITERATURE SURVEY	3
3.	RESEARCH GAPS OF EXISTING METHODS	13
	3.1 Limited Dataset and Variability	13
	3.2 Challenges in Gesture Recognition	13
	3.3 High Computational Complexity and Real Time Limitations	13
	3.4 Poor Integration of ISL Grammar	14
	3.5 Inconsistent Text-to-Speech (TTS) Conversion	14
	3.6 Limited Deployment in Real World Scenarios	14
4.	PROPOSED METHODOLOGY	15
	4.1 Database Collection / Image Acquisition	15
	4.2 Image Preprocessing	16
	4.3 Feature Extraction	17
	4.4 Classification	18
5.	OBJECTIVES	19
	5.1 Design an AI Based System for Accurate Recognition	19
	5.2 Translate Recognized ISL Gestures into Real Time Text and Speech Output	19
	5.3 Provide a Communication Tool for the Deaf and Hard of Hearing Community	19
	5.4 Optimize Gesture Recognition Algorithms for Real World Variability	20
	5.5 Develop an Intuitive, User Friendly Application or Device for Broad Adoption	20
6.	SYSTEM DESIGN & IMPLEMENTATION	21
	6.1 System Architecture Overview	21
	6.2 Implementation Details	22
	6.2.1 Image Acquisition and Preprocessing	22
	6.2.2 Model Design and Training	22
	6.2.3 Gesture Classification and Text Generation	23
	6.2.4 Text-to-Speech Conversion	23

	6.2.5 Real Time Integration	23
7.	TIMELINE FOR EXECUTION OF PROJECT	24
8.	OUTCOMES	25
	8.1 Enables Real Time Interaction	25
	8.2 Reducing Communication Barriers	25
	8.3 Empowering Individuals with Hearing Impairments	25
	8.4 Encourages Inclusivity in Digital Platforms, Allowing Sign	26
	8.5 Improves Accessibility in Healthcare, Education, and Public Services	26
9.	RESULTS AND DISCUSSIONS	27
10.	CONCLUSION	29
	REFERENCES	30
	APPENDIX-A [PSUEDOCODE]	32
	APPENDIX-B [SCREENSHOTS]	41
	APPENDIX-C [ENCLOSURES]	43
	SUSTAINABLE DEVELOPMENT GOALS	44

Chapter 1

INTRODUCTION

Sign language is a visual gestural language used by the deaf and hard of hearing community to communicate. Unlike spoken languages, it relies on hand signs, facial expressions, and body movements to convey meaning. **Indian Sign Language (ISL)** is the mode of communication for the deaf community in India, with its own unique gestures and grammar. However, due to limited awareness and resources, ISL remains largely inaccessible to the hearing population, creating a communication barrier. While full word sign recognition is complex, finger spelling where words are spelled letter by letter provides a structured way to interpret ISL into text and speech. Several researchers have explored sign language recognition using different approaches. Some studies focus on vision based techniques using deep learning models like **Convolutional Neural Networks (CNNs)** and **Recurrent Neural Networks (RNNs)** to recognize dynamic gestures, while others have used sensor based gloves or motion capture devices for improved accuracy. In the context of ISL, previous works have attempted to recognize isolated signs or alphabets, but real time translation systems remain limited. Most existing solutions either require expensive hardware or struggle with generalization across different users and lighting conditions.

In this project, we focus on recognizing ISL alphabets in real time using a machine learning based approach. By using a pre trained model on a dataset of hand gestures corresponding to each letter in the ISL alphabet and leveraging computer vision techniques and deep learning architectures, our system captures hand movements through a camera, processes the images, and classifies them into the correct alphabets. The recognized letters are then combined to form words, which can be converted into speech using Text-to-Speech synthesis. This approach simplifies the recognition process by breaking down communication into individual letters rather than full word signs. One of the key challenges in ISL recognition is handling variations in hand shapes, lighting conditions, and background noise. To improve accuracy, we implemented image pre processing techniques such as rotation and horizontal flipping to accommodate mirrored and varied hand orientations. We also simulated different lighting conditions and image blurriness to help the model adapt to real world scenarios.

Moreover, this project aligns with the broader goal of promoting inclusivity and digital accessibility for the Deaf and hard of hearing communities. As AI driven technologies become

more prevalent, there is a growing need for scalable, low cost solutions that bridge communication gaps. Our system is designed to run on standard consumer grade hardware, making it affordable and easy to deploy across educational institutions, public services, and personal settings. By focusing on ISL a language that is often underrepresented in global research this work contributes to the diversification of AI applications and supports the inclusion of Indian linguistic and cultural contexts in emerging technologies. Future enhancements may include integrating **Natural Language Processing** (NLP) for better word formation and expanding the system to recognize common ISL words and phrases. By refining our model and making it more robust, we aim to create a practical and accessible communication tool for the Deaf community in India.

Chapter 2

LITERATURE SURVEY

Table 2.1: Study of Tools and Methodologies

References No	Year	Study of Tools/Technology	Overall Accuracy	Dataset
[1]	2022	SURF for feature extraction, BOVW for visual word mapping, SVM (99.17%) and CNN (99.64%) for classification, Pyttsx3 for text to speech, Google Speech API for speech to text.	Achieved 99.17% with SVM and 99.64% with CNN by combining SURF feature extraction with BOVW for robust visual word mapping, improving recognition accuracy across diverse ISL signs.	Custom dataset with 36,000 images (26 alphabets + 10 numerals), collected using webcam with two image acquisition methods (plain background and background subtraction).
[2]	2015	Otsu's algorithm for segmentation, SIFT & HOG for feature extraction, ANN for classification, and PCA for improved accuracy.	Achieved 93% accuracy by combining Otsu's segmentation with SIFT and HOG for robust feature extraction, and ANN for effective classification.	Finger spelling dataset for Indian Sign Language (ISL).

[3]	2007	HMM for dynamic gesture recognition, Particle Filtering for motion tracking, FSM for modeling gesture sequences, ANN for facial and hand gesture recognition, PCA for dimensionality reduction.	Achieved 96.21% for gesture recognition by combining HMM for sequential pattern recognition and ANN for classifying hand motion trajectories.	ASL (American Sign Language) gesture dataset with 40 complex hand gestures and approximately 38 instances per sign.
[4]	2023	Progressive transfer learning with S3D for visual encoding, mBART for text generation, and V L Mapper for bridging visual and language modalities.	Achieved 28.39 BLEU 4 score by combining progressive pretraining on general and within domain datasets, boosting performance with improved visual and language representation learning.	PHOENIX 2014T (German Sign Language) and CSL Daily (Chinese Sign Language) datasets.
[5]	2013	HSI color model for segmentation, Distance Transform for centroid detection, and Back Propagation Neural	Achieved 99% accuracy using BPN by combining effective segmentation with	Custom dataset with ISL alphabets captured through

		Network (BPN) with 2 hidden layers for classification.	optimized neural network architecture under good illumination conditions.	webcam with 80 training videos and 70 testing videos.
[6]	2020	Maya for 3D animation, Adobe Photoshop for graphic design, Java for Android app development, and PSL (Pakistan Sign Language) for gesture representation.	Achieved 95% accuracy by integrating Maya based 3D animations with real time text to animation conversion for Urdu, Sindhi, and English sign language gestures.	Custom built database containing PSL gestures for alphabets, numbers, and common phrases.
[7]	2020	CNN for feature extraction and classification, Image Augmentation for improving model robustness, and Google Text-to-Speech (GTTS) for converting recognized Arabic signs into speech.	Achieved 90% accuracy by optimizing convolution layers with 32 and 64 kernels, dropout rates of 0.25 and 0.5, and enhancing performance through image augmentation.	Custom dataset with 125 images per letter for 31 Arabic Sign Language (ArSL) letters, split 80:20 for training and testing.
[8]	2004	HMM based framework for ASL recognition using the Movement Hold model for phoneme representation, with multiple independent channels for simultaneous event recognition.	Achieved 95% accuracy by integrating handshape recognition and independent channel modeling to reduce	ASL dataset with a 22 sign vocabulary designed for testing simultaneous gesture events.

			computational complexity.	
[9]	2023	Deep transfer learning with VGG16, VGG19, InceptionV3, Xception, and ResNet50 as backbone networks combined with a Random Forest classifier for Bangla Sign Language recognition.	Achieved 91.67% for character recognition and 97.33% for digit recognition using background elimination, transfer learning, and optimal batch size tuning.	Ishara Bochon (digit dataset) and Ishara Lipi (character dataset) for Bangla Sign Language.
[10]	2020	HMM for sequence modeling, CRF for context based labeling, DTW for aligning temporal sign data, and CNN BLSTM for deep learning based gesture recognition.	Achieved 85% segmentation accuracy using CRF for segmenting sign boundaries and 83% recognition accuracy with CNN BLSTM by combining visual feature extraction with sequential modeling.	RWTH PHOENIX Weather 2014 and SIGNUM datasets for German Sign Language recognition.
[11]	2023	CNN for feature extraction and classification, augmented dataset with affine transformations to improve generalization, and a custom ISL dataset collected from 65 users in uncontrolled environments.	Achieved 92.43%, 88.01%, and 99.52% accuracy across three datasets by combining CNN with data augmentation for improved robustness and performance.	Custom ISL dataset (65 users), self collected ISL dataset, and publicly available ASL dataset.

[12]	2003	Real time hand tracking with motion detection, skin color extraction, and edge detection; FD for spatial features; Optical Flow for motion analysis; HMM for dynamic gesture recognition.	Achieved 93.5% accuracy by combining FD for shape features and Optical Flow for motion features, improving recognition of complex dynamic gestures.	Custom dataset with 20 different hand gestures, each performed 3 times by 20 individuals (total 1200 sequences).
[13]	2014	Haar like feature based cascaded classifier for hand detection, HSV color model for skin color segmentation, and KNN classifier for Bengali Sign Language recognition.	Achieved 98.17% for vowels and 94.75% for consonants by combining Haar like feature detection with skin color segmentation and optimized KNN classifier settings.	Custom dataset with 3600 images for training and 3600 images for testing (10 participants performing 36 signs each in varying conditions).
[14]	2006	Skin detection using color, motion, and position cues; Kalman filter for occlusion detection; and blob matching for tracking face and hands.	Achieved 93.5% accuracy with Kalman filter based occlusion handling and integrated segmentation tracking for improved robustness.	ECHO database with 600 labeled frames, including 237 occluded frames for evaluation.

[15]	2008	Lexical similarity analysis, Recorded Text Testing (RTT) for dialect intelligibility, and language attitude assessment to analyze ISL varieties and prioritize literature development.	Achieved high reliability in identifying ISL dialect patterns by merging data from RTT, lexical analysis, and self reported language ability for improved linguistic assessment.	Indian Sign Language (ISL) dataset collected from five major Indian cities (Mumbai, Delhi, Hyderabad, Chennai, and Kolkata).
[16]	2021	Machine translation approaches including rule based, statistical, and neural networks; Avatar technology for gesture generation; and sign language repositories for gesture datasets.	Achieved high accuracy by combining neural networks for improved translation precision and avatar based gesture generation for enhanced visual representation.	Multiple datasets including RWTH PHOENIX Weather 2014, eSign 3D, and custom repositories for various sign languages.

[1] The research paper presents a system for recognizing Indian Sign Language (ISL) using a combination of Speeded Up Robust Features (SURF), Support Vector Machine (SVM), and Convolutional Neural Networks (CNN). The system captures hand signs from a live video feed, processes them through feature extraction and classification, and outputs the recognized symbols as text and speech. The model achieves over 99% accuracy and includes a graphical interface for user interaction. The approach improves ISL recognition by addressing background dependency and supporting both single and double handed signs.

[2] This research paper analyzes recent advancements in Indian Sign Language (ISL) recognition, highlighting its complexity compared to American Sign Language (ASL). It reviews various approaches, including sensor based and vision based techniques, and discusses challenges such as gesture similarity, dynamic gestures, and facial expressions. The study emphasizes the need to evaluate different feature extraction and classification methods for ISL's unique complexities. It concludes that while global sign language recognition has advanced, ISL requires specialized approaches for higher accuracy and real world usability.

[3] This research paper provides a comprehensive survey on gesture recognition, emphasizing hand gestures and facial expressions in human computer interaction. It explores various methods, including Hidden Markov Models (HMMs), particle filtering, finite state machines, and artificial neural networks. The study discusses applications in sign language recognition, medical rehabilitation, virtual reality, and human computer interfaces. It highlights challenges such as variability in gestures, environmental factors, and computational complexity, suggesting hybrid approaches combining statistical models and soft computing for improved accuracy and robustness in gesture recognition systems.

[4] This research paper presents a multi modality transfer learning approach for sign language translation. It addresses data scarcity by progressively pretraining a model in two stages: first on general domain human actions and multilingual text, then on sign language specific datasets. A visual language mapper bridges visual and text based models for end to end learning. The approach outperforms previous state of the art methods on benchmarks like PHOENIX 2014T and CSL Daily, demonstrating the effectiveness of transfer learning in improving sign language translation accuracy.

[5] This research paper presents a neural network based approach for recognizing Indian Sign Language (ISL) characters. It captures hand gestures via a webcam, extracts features such as finger angles and positions, and uses a backpropagation neural network to classify them into alphabets. The study compares different neural network architectures, highlighting improvements in accuracy using segmentation and feature extraction techniques. The proposed model achieves high accuracy under good lighting conditions, demonstrating its potential for real time ISL character recognition without additional hardware like gloves or markers.

[6] This research paper presents a text to animation system for Urdu and Sindhi sign languages, aiming to bridge communication gaps between deaf/mute individuals and the general public. The system converts text into animated hand gestures using a virtual avatar, allowing users to learn and communicate through an Android app. It includes tutorials for alphabets, numbers, and basic sentences in English, Urdu, and Sindhi. The application enhances accessibility, making sign language education more interactive and widely available.

[7] This research paper presents a Convolutional Neural Network (CNN) based system for Arabic Sign Language (ArSL) recognition and speech generation. The model detects hand signs from images, classifies them into Arabic letters, and converts them into spoken Arabic using Text-to-Speech synthesis. It achieves 90% accuracy, with improvements from data augmentation. The system aims to aid communication for the hearing impaired, with potential enhancements through advanced sensors like Leap Motion or Kinect. Future work includes expanding datasets for better recognition accuracy.

[8] This research paper presents a framework for recognizing American Sign Language (ASL) using a multi channel approach. It addresses challenges in modeling simultaneous hand movements and handshape changes, which create complexity in ASL recognition. The study integrates the Movement Hold phonological model and uses Hidden Markov Models (HMMs) to break down signs into phonemes, improving recognition efficiency. Experimental validation with a 22 sign vocabulary demonstrates the benefits of independent channels for movement and handshape, making ASL recognition more computationally feasible while maintaining accuracy.

[9] This research paper presents a hybrid deep learning approach for Bangla Sign Language (BSL) recognition, combining a transfer learning based Convolutional Neural Network (CNN) with a Random Forest (RF) classifier. The model achieves high accuracy for both character and digit recognition using the Ishara Bochon and Ishara Lipi datasets. It also incorporates a background elimination algorithm to improve feature extraction. Experimental results demonstrate accuracy, precision, recall, and F1 score improvements over existing methods, making the system effective for automatic BSL recognition.

[10] This research paper reviews vision based continuous sign language recognition (CSLR)

systems, highlighting challenges like movement epenthesis, signer variations, and complex background segmentation. Traditional methods like Hidden Markov Models (HMMs), Conditional Random Fields (CRFs), and Dynamic Time Warping (DTW) have been used but face limitations. The study explores deep learning based approaches, including CNN LSTM models, which improve recognition accuracy without explicit segmentation. It concludes that CSLR still requires advancements in handling real world data, large vocabularies, and multimodal feature integration.

[11] This research paper provides a comprehensive review of vision based Indian Sign Language (ISL) recognition systems. It discusses different approaches, including digital image processing, machine learning, and deep learning techniques, analyzing their effectiveness in gesture recognition. The study highlights challenges such as dataset limitations, background variations, and signer dependent recognition. It also presents future research directions, emphasizing the need for large benchmark datasets, improved recognition of dynamic and continuous gestures, and advanced deep learning models to enhance ISL recognition accuracy.

[12] This research paper introduces a real time hand gesture recognition system using Hidden Markov Models (HMMs). The system consists of four modules: real time hand tracking, feature extraction, HMM training, and gesture recognition. It employs Fourier descriptors for spatial features and motion analysis for temporal features, integrating them into a feature vector. The model achieves over 90% accuracy in recognizing 20 different gestures. The approach eliminates the need for gloves or markers, ensuring reliable recognition in complex backgrounds with minimal computational overhead.

[13] The paper presents a real time Bengali Sign Language (BdSL) recognition system using computer vision. It detects hand gestures with Haar like classifiers, extracts signs via skin color segmentation, and classifies them using a K Nearest Neighbors (KNN) model. Trained on 3600 images, the system achieved 98.17% accuracy for vowels and 94.75% for consonants. While challenges like background interference and visually similar signs exist, the system demonstrates high accuracy and real time performance, enhancing communication accessibility for the Bengali deaf community.

[14] This research paper presents a unified system for segmenting and tracking face and hand movements in sign language recognition. Unlike previous approaches that use colored gloves,

it combines color, motion, and position features for skin detection. A Kalman filter based algorithm is employed to track hands and handle occlusions between them and the face. The system improves segmentation accuracy by integrating tracking information, achieving low error rates in real world sign language conversations. Experimental results confirm its effectiveness in natural sign language video sequences.

[15] This research paper examines the regional variations, vitality, and identity of Indian Sign Language (ISL) across five cities in India. Using lexical similarity analysis, dialect intelligibility testing, and language attitude assessments, the study concludes that ISL consists of multiple dialects rather than separate languages. Mumbai's dialect is identified as the most prestigious and widely understood, making it ideal for initial literature development. The paper also introduces a novel data merging approach for sign language diversity assessment and suggests further research to support ISL standardization and recognition.

[16] This research paper provides a systematic review of machine translation approaches for sign language, covering natural language to sign language translation, gesture recognition, and avatar based sign generation. It analyzes 147 research articles and categorizes existing methods, including rule based, statistical, and deep learning approaches. The study highlights key challenges such as dataset limitations, linguistic differences, and computational constraints. It concludes with future directions for improving sign language translation using neural networks and larger annotated datasets to enhance accessibility for the deaf community.

Chapter 3

RESEARCH GAPS OF EXISTING METHODS

3.1 Limited Dataset and Variability

- One of the major challenges in Indian Sign Language (ISL) recognition is the lack of large, diverse, and standardized datasets.
- Without standardized ISL datasets that capture a wide range of real world conditions, training robust and reliable AI models remains a difficult task.
- Most existing models are trained on relatively small datasets, which significantly limits their ability to generalize across different users. Variations in individual signing styles, hand sizes, and orientations often lead to reduced recognition accuracy.

3.2 Challenges in Gesture Recognition

- Gesture recognition, particularly for sign languages, involves detecting subtle hand movements and sometimes facial expressions. While many models demonstrate high accuracy for static signs (such as isolated alphabets), they often struggle with dynamic gestures that involve motion and temporal changes.
- The presence of complex backgrounds can interfere with gesture detection, making models unreliable in uncontrolled environments.
- Although wearable sensor based solutions can enhance accuracy, they are often uncomfortable for users and can be prohibitively expensive for widespread adoption.

3.3 High Computational Complexity and Real Time Limitations

- Deep learning models used for computer vision based ISL recognition typically require significant computational resources, which can be a barrier for real time deployment on mobile or low power devices.
- High model complexity often translates to increased latency, hindering the feasibility of live communication. As a result, many existing systems are unable to deliver smooth, real time performance, especially on consumer grade hardware, limiting their practical usability.

3.4 Poor Integration of ISL Grammar

- ISL has its own unique grammatical structure, which differs significantly from spoken and written languages. Many existing recognition systems fail to incorporate this grammatical nuance, leading to incorrect or incomplete translations.
- Most solutions only map recognized signs to individual words without considering sentence structure, resulting in translations that are not correct and lack clarity.
- Furthermore, the absence of context aware Natural Language Processing (NLP) models contributes to poor synthesis of grammatically correct and contextually appropriate speech.

3.5 Inconsistent Text-to-Speech (TTS) Conversion

- Many systems are capable of accurately converting gestures to text, and then to a Text-to-Speech (TTS) sentence which often lacks fluency.
- The generated speech may sound robotic or unnatural, diminishing the effectiveness of the communication.
- Additionally, limited support for multiple Indian languages further restricts accessibility, especially in a linguistically diverse country like India. This reduces the system's usability among non English speaking users.

3.6 Limited Deployment in Real World Scenarios

- Despite promising results in lab settings, many ISL recognition systems fail to perform in real world scenarios. These models often rely on controlled conditions such as uniform lighting and plain backgrounds, which are not always feasible in practical environments.
- The lack of training on diverse user groups makes these systems poorly adaptable to different users, especially when factors like skin tone, hand shape, and regional variations in signing come into play. This limits the broader applicability and user friendliness of such technologies.

Chapter 4

PROPOSED METHODOLOGY

4.1 Database Collection / Image Acquisition

The dataset for this research consists of images representing different hand signs used in sign language communication. The images were collected through a structured approach, either by capturing them manually using a webcam or sourcing them from existing sign language datasets. Manual image acquisition was performed (Fig 4.1) in a controlled environment to ensure uniformity while also incorporating variations in lighting, backgrounds, and angles to introduce real world diversity. The primary objective during this phase was to ensure that the dataset was sufficiently large and diverse to allow the deep learning model to generalize well across different hand shapes, skin tones, and lighting conditions. To maintain organization and facilitate efficient data processing, the dataset was categorized into separate directories, with each directory representing a unique sign. This structured format allowed for easy loading and pre processing using deep learning frameworks. One of the key challenges during data collection was ensuring that the dataset was balanced, meaning each class had a sufficient number of samples to prevent model bias. To address this, an equal number of images were collected per class, and additional images were generated using augmentation techniques. Furthermore, background noise interference was a major concern, as hand gestures needed to be clearly distinguished from the surroundings. To mitigate this, images were captured against varied backgrounds, including solid colored backdrops and cluttered environments, ensuring that the model learned to focus on the hand rather than the background elements. To enhance the dataset further, an additional feature extraction step was implemented using Media pipe Hands, a real time hand tracking framework that detects 21 key landmarks on the hand as seen in Fig 4.2. This generated hand landmark data, which was stored in hand landmarks.csv and hand landmarks.json formats. These files contained the precise coordinates of the detected hand key points, which could be used later for hybrid models combining convolutional neural networks (CNNs) with recurrent models such as long short term memory (LSTM) networks. This alternative approach could potentially improve model accuracy by relying on skeletal hand structures rather than raw pixel data alone.



Fig. 4.1. ISL signs captured through a webcam.

4.2 Image Pre Processing

Pre processing plays a crucial role in preparing the collected images for effective model training. Since raw images often contain noise, inconsistencies, and variations in scale, applying a set of transformations ensures that the model learns meaningful features rather than being affected by irrelevant details. The first step in this process involved resizing all images to a fixed dimension of 224x224 pixels, aligning with the input size required by MobileNetV2. This step ensures uniformity across the dataset, as neural networks perform best when the input dimensions are consistent. To improve model convergence, normalization was applied by scaling pixel values to the range [0,1]. This step prevents large numerical values from dominating smaller ones during back propagation, thereby stabilizing the training process. A significant aspect of pre processing was data augmentation, implemented using Augmentations, a powerful image augmentation library. Augmentation techniques were strategically chosen to enhance model generalization by introducing controlled variations in the dataset. Rotation ($\pm 15^\circ$) was applied to account for different hand orientations, ensuring that the model learned rotational invariance. Horizontal flipping was included to help the model recognize mirrored gestures, a critical aspect in sign language recognition where left handed and right handed gestures must be correctly classified. In addition to these basic augmentations, more advanced transformations were applied to improve robustness against varying environmental conditions. Brightness and contrast adjustments helped simulate different lighting conditions, ensuring that the model could recognize hand signs under both bright and dim lighting. Gaussian noise was added to introduce real world imperfections, mimicking conditions where images might be slightly blurry due to camera limitations. Furthermore, CLAHE (Contrast Limited Adaptive Histogram Equalization) was applied to enhance contrast in images where hand visibility was poor, ensuring that hand features remained prominent. The combination of these augmentations significantly enriched the dataset, preventing over fitting while enhancing the model's ability to generalize to unseen data. An additional pre processing step was the use of hand landmarks, extracted using Mediapipe. The 21 detected key points were saved in structured formats (CSV and JSON), enabling future models to incorporate landmark based recognition, potentially improving accuracy by shifting the focus from raw

pixel data to skeletal hand movement patterns. This multi modal approach opens possibilities for combining CNN based image classification with LSTM based temporal tracking of hand movements.

4.3 Feature Extraction

Feature extraction is the process of identifying significant visual patterns within images that help distinguish different hand signs. Instead of training a model from scratch, which requires massive datasets and computational power, transfer learning was employed using MobileNetV2, a state of the art deep learning model pre trained on the ImageNet dataset. MobileNetV2 was chosen for its efficiency, lightweight architecture, and high accuracy, making it suitable for real time applications such as sign language recognition. MobileNetV2 operates using depthwise separable convolutions, reducing computational complexity while maintaining high accuracy. The lower layers of the model extract basic features such as edges, curves, and textures, while the higher layers capture complex features such as shapes and object structures. Since MobileNetV2 was trained on a large scale dataset of real world images, its lower layers contained useful general purpose features, which were retained during training. Only the higher level layers were fine tuned to specialize in sign language recognition. The extracted features were passed through a Global Average Pooling layer, which condenses feature maps into a single vector, making the model more efficient while reducing over fitting. To further refine the extracted features, two fully connected layers were added: one with 256 neurons followed by another with 128 neurons, both using the ReLU activation function to introduce non linearity. To prevent overfitting, dropout regularization was applied at rates of 0.5 and 0.4, respectively, randomly deactivating a fraction of neurons during training to enhance generalization. An alternative feature extraction approach considered for future work involves using hand landmarks instead of full images. By utilizing the 21 key points detected by Mediapipe, a model could learn gesture patterns based on skeletal structures rather than pixel based representations. This method could improve classification efficiency, particularly in scenarios with background noise or occlusions. However, for this research, the focus remains on CNN based feature extraction using MobileNetV2.



Fig.4.2. Hand Landmarks on a captured image

4.4 Classification

The classification process involves mapping extracted features to specific hand signs using a neural network. The final output layer of the model consists of N neurons, corresponding to the number of unique hand signs in the dataset, with a softmax activation function to produce probability distributions over possible classes. The model was compiled using the Adam optimizer with an initial learning rate of 0.0001, which was later reduced during fine tuning. The categorical cross entropy loss function was used, since this is a multiclass classification problem. To improve training efficiency, two critical callbacks were implemented: EarlyStopping and ReduceLROnPlateau. EarlyStopping monitored the validation loss and stopped training if no improvement was observed for five consecutive epochs, preventing unnecessary computations and reducing overfitting risks. ReduceLROnPlateau dynamically adjusted the learning rate, reducing it by a factor of 0.2 if the validation loss plateaued for 3 epochs, ensuring that the model continued to progress during later stages of training. After initial training, a fine tuning phase was performed. During this phase, the deeper layers of MobileNetV2 were unfrozen, allowing the model to adjust high level feature representations for sign language recognition. To avoid catastrophic forgetting, only the last few layers were made trainable, while the initial layers remained frozen. This fine tuning phase was trained for an additional 10 epochs with a lower learning rate of 0.00001, refining the model's ability to distinguish subtle variations between hand signs. The trained model was evaluated using accuracy and loss metrics, along with a confusion matrix to analyze misclassifications. The expected accuracy target for this research was 85% or higher, with an emphasis on optimizing the model for real time inference. Using MobileNetV2, the trained model achieved efficient classification with an inference time of approximately 10 milliseconds per frame on the GPU, making it suitable for practical sign language applications.

Chapter 5

OBJECTIVES

5.1. Design an AI Based System for Accurate Recognition of Indian Sign Language (ISL) Gestures:

- The primary objective of this research is to design and develop an artificial intelligence powered system capable of recognizing static Indian Sign Language (ISL) gestures with high accuracy.
- This involves the integration of **computer vision** and **deep learning** techniques, particularly CNNs, to analyse visual input and identify hand gestures corresponding to letters of the ISL alphabet. The model is expected to generalize well across different users, hand sizes, and background conditions.

5.2. Translate Recognized ISL Gestures into Real Time Text and Speech Output:

- Once the hand gesture is accurately classified, the system translates the recognized gesture into readable text. To facilitate effective communication between ISL users and individuals unfamiliar with sign language, this text is further converted into **natural sounding speech** using a Text-to-Speech (TTS) engine.
- The goal is to achieve a **real time, end to end communication bridge** that allows ISL users to express themselves instantly and clearly in spoken language.

5.3. Provide a Communication Tool for the Deaf and Hard of Hearing Community:

- This system is intended to serve as a **practical assistive technology** for the deaf and hard of hearing community. By enabling gesture based input and audio output, it empowers users to communicate confidently in **educational, workplace, healthcare, and social environments**.
- The project aims to reduce dependency on human interpreters and bridge communication gaps in both public and private settings.

5.4. Optimize Gesture Recognition Algorithms for Real World Variability:

- A critical objective is to ensure the robustness of the system under varying conditions such as different hand orientations, lighting environments, and backgrounds. This involves enhancing the gesture recognition algorithm's ability to **accurately interpret gestures despite changes in scale, angle, and illumination** which is a common challenge in real world applications.

5.5. Develop an Intuitive, User Friendly Application or Device for Broad Adoption:

- The final goal is to encapsulate the technology into a **simple, accessible, and user friendly application or device** that can be readily adopted by users with minimal training. The system should require no special hardware beyond a standard camera and should offer real time responsiveness.
- Usability and accessibility are key design considerations, ensuring that the system can be deployed in **educational institutions, public service points, and even personal devices** with ease.

Chapter 6

SYSTEM DESIGN & IMPLEMENTATION

The proposed system is designed as a modular pipeline that converts Indian Sign Language (ISL) hand gestures into audible speech. The system comprises five main stages: data acquisition, preprocessing, gesture recognition using a convolutional neural network, text generation, and speech synthesis. Each stage is implemented using open source tools and frameworks to ensure reproducibility, scalability, and real time performance.

6.1. System Architecture Overview

The system architecture follows a sequential flow:

1. **Input Capture:** Live hand gesture input is captured via a webcam or imported from image files.
2. **Preprocessing Module:** Images are resized, normalized, and optionally augmented to prepare for classification.
3. **Gesture Recognition Model:** A CNN model based on ResNet50 processes the image and outputs the predicted gesture class.
4. **Text Mapping Layer:** The predicted class index is mapped to its corresponding alphabet or symbol.
5. **Speech Synthesis:** The mapped text is converted to speech using a Text-to-Speech (TTS) engine.

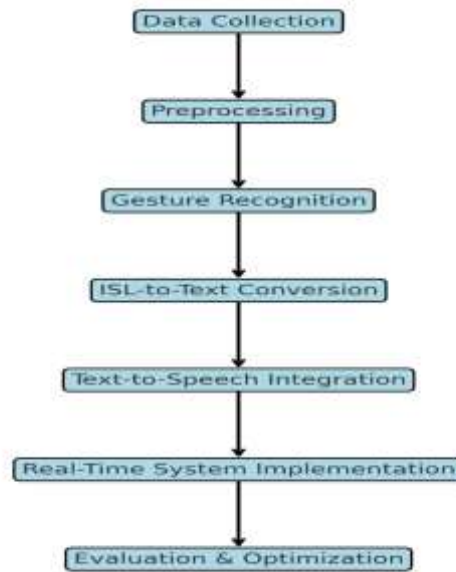


Fig 6. 1 System Design Diagram

This modular structure(Fig 6.1) ensures clarity in implementation and facilitates future upgrades (e.g., adding dynamic gesture support or multilingual speech synthesis).

6.2. Implementation Details

6.2.1. Image Acquisition and Preprocessing

- The system uses **OpenCV** to interface with the webcam and capture gesture images in real time.
- Each image is resized to **224×224 pixels** to match the model's input dimensions.
- Pixel values are normalized to the [0, 1] range to improve convergence during training.
- **ImageDataGenerator** from Keras is used to apply data augmentation (rotation, zoom, shear, and horizontal flip).

6.2.2. Model Design and Training

- A MobileNetV2 architecture is employed as the base model with pre trained ImageNet weights.
- The top layers are replaced with custom dense layers, batch normalization, and dropout to improve generalization.

- The model is compiled using the **Adam optimizer** with categorical crossentropy as the loss function.
- Training is performed on a labeled dataset of 35 ISL classes using a batch size of 32 for 25 epochs.
- The model achieved a precision of **99%** indicating strong learning with minimal overfitting.

6.2.3. Gesture Classification and Text Generation

- After prediction, the model outputs a class index corresponding to an ISL alphabet.
- A mapping function converts this index into the appropriate textual character .

6.2.4. Text-to-Speech Conversion

- The system uses **Google Text-to-Speech (gTTS)** to convert the predicted text into an audio file.
- The audio is automatically played back using a media playback library .

6.2.5. Real Time Integration

- A real time prediction loop is implemented to continuously capture frames, classify gestures, and output speech.
- The UI displays the predicted letter on the video feed, providing instant visual feedback alongside the audio.

Chapter 7

TIMELINE FOR EXECUTION OF PROJECT (GANTT CHART)

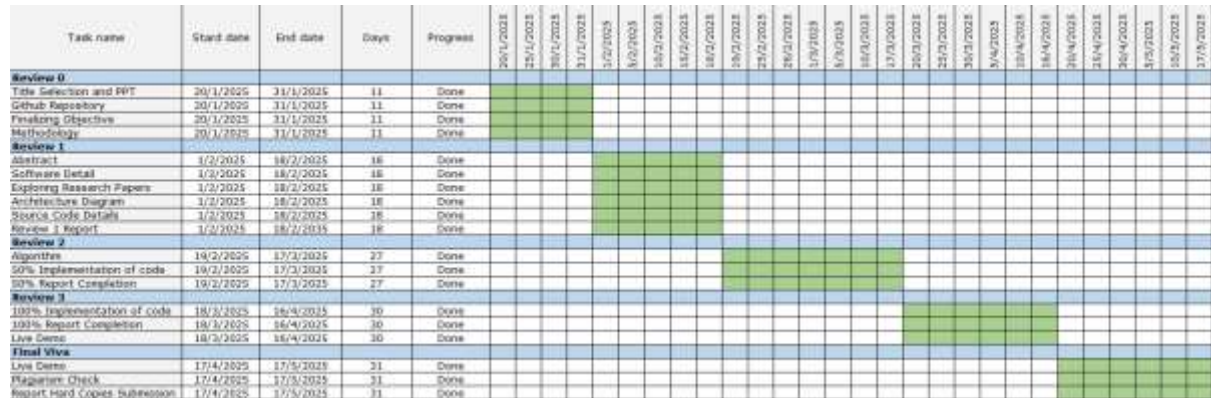


Fig. 7.1. Time line of the project

We can see in Figure 7.1 that the Gantt chart outlines the project timeline across five key review phases, beginning in early February 2025 and concluding with the final viva voce in Mid May 2025. Each review stage is aligned with specific tasks, enabling structured progress tracking from initial framework design and simulation setup to data collection, model development, and reporting. This visual representation helps in understanding the sequential flow and overlapping of activities, ensuring timely completion of milestones and preparation for evaluations.

Chapter 8

OUTCOMES

8.1. Enables Real Time Interaction Between ISL Users and Non Sign Language Users:

- The system facilitates seamless, real time communication between both by converting static hand gestures into spoken words, the model bridges the language gap instantly, reducing the need for human interpreters.
- This promotes more natural, spontaneous conversations in both personal and public settings, significantly improving social integration.

8.2. Reduces Communication Barriers in Education, Workplaces, and Public Services:

- Communication challenges faced by hearing impaired individuals in classrooms, offices, and public institutions often hinder their access to critical information and opportunities. The proposed system addresses this gap by translating sign language into speech, enabling inclusive participation in lectures, meetings, customer service interactions, and official procedures.
- It contributes to developing a more equitable environment where accessibility is not an afterthought but a built in feature.

8.3. Empowers Individuals with Hearing Impairments by Giving Them a Voice in Mainstream Conversations:

- This technology empowers users by transforming their hand gestures into audible language, thereby "giving a voice" to those who are otherwise non verbal. It enables users to express themselves confidently and independently in group discussions, social gatherings, or professional engagements.
- This autonomy not only builds self esteem but also strengthens their presence and involvement in everyday societal activities.

8.4. Encourages Inclusivity in Digital Platforms, Allowing Sign Language Users to Engage in Online Content More Effectively:

- As digital communication becomes the norm, inclusivity in online platforms is more important than ever. The proposed system can be integrated into video conferencing tools, virtual classrooms, and social media to allow sign language users to communicate effectively and participate fully.
- It supports the creation of more accessible digital experiences, closing the gap between differently abled users and the broader digital audience.

8.5. Improves Accessibility in Healthcare, Education, and Public Services Through Automated ISL Translation:

- Access to healthcare, education, and government services is a fundamental right, yet communication barriers often exclude hearing impaired individuals from fully utilizing these services.
- By providing automated ISL to speech translation, the system can assist healthcare professionals in understanding patients' needs, support educators in delivering content, and help government officials communicate policies and instructions ensuring that no one is left behind due to language limitations.

Chapter 9

RESULTS AND DISCUSSIONS

The model predicts the class of a given image by processing the input through its convolutional layers and generating a probability distribution over all the classes. The predicted class is the one with the highest probability.

- **Softmax Output:** The model produces a probability score for each class using the softmax activation function. This allows us to interpret the model's confidence in its prediction.
- **Top k Predictions:** In cases where misclassification occurs, analyzing the top 3 predictions can help understand confusion between similar signs.
- **Threshold Based Filtering:** To improve reliability in real time applications, a confidence threshold (e.g., 80% or higher) can be applied to reduce uncertain predictions.

The training process demonstrated rapid convergence, with the model reaching a validation accuracy of 99% by the final epoch. The initial training epochs showed a significant improvement in accuracy, stabilizing in later epochs. The final loss and accuracy values indicate that the model effectively learned the representations needed for accurate classification of sign language gestures. The classification report provided a detailed breakdown of performance across all sign categories, including precision, recall, and F1 score. The results indicate that:

- **Precision:** The model achieved an average precision of over 99%, indicating that incorrect predictions were minimal.
- **Recall:** The recall values were consistently high, meaning the model was able to correctly identify almost all instances of each class.
- **F1 score:** The F1 scores were close to 1.0 across all classes, demonstrating a strong balance between precision and recall.

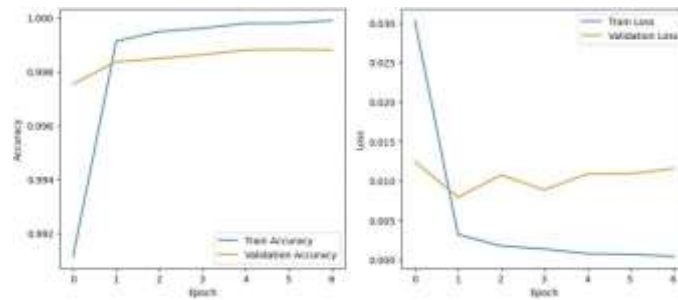


Fig. 9.1. Performance of the model

As we can see from Fig 9.1, the model responds well to unseen data, as reflected in the high validation accuracy. The minor classification errors can be further improved through data augmentation and model fine tuning. The low validation loss indicates that the model is not overfitting and maintains robust performance on test data. To evaluate the model's effectiveness in practical scenarios, we tested it in a real time environment using a webcam based sign recognition system. The model was integrated into a real time prediction pipeline, where it processed live hand gesture inputs and classified them into corresponding sign language characters (A Z, 1 9). The MobileNetV2 based model demonstrated fast inference speeds and maintained high accuracy. The average prediction time per frame was less than 50 milliseconds, ensuring minimal latency. However, certain challenges, such as lighting variations, hand positioning, and occlusions, occasionally led to misclassifications.

Chapter 10

CONCLUSION

This research presents a comprehensive and efficient approach to real time sign language recognition using deep learning techniques, with a particular focus on the Indian Sign Language (ISL). The combination of a well structured and diverse dataset, robust image pre processing techniques, and the use of transfer learning with MobileNetV2 contributed to achieving high classification accuracy and rapid inference times. The classification model achieved an impressive validation accuracy of 99%, with precision, recall, and F1 scores nearing perfection across all classes. Augmentation techniques played a vital role in ensuring generalization and reducing overfitting, while callbacks like Early Stopping and learning rate scheduling improved training efficiency. The model also demonstrated strong performance in real time scenarios, achieving inference speeds suitable for practical deployment.

While challenges such as background noise, lighting inconsistencies, and hand occlusions were partially mitigated through data collection and augmentation strategies, future work could further address these issues by enhancing landmark based recognition or incorporating temporal dynamics through recurrent neural networks. While minor challenges exist, the overall performance suggests that the model is well suited for deployment in assistive applications, bridging the communication gap for individuals with hearing or speech impairments.

The system can be useful for ISL numeral and alphabet signs only. It cannot be considered as a complete system, as for complete recognition of sign language we also have to include sentences. We plan to integrate model with NLP techniques to construct meaningful sentences from recognized gestures, enhancing its usability in communication systems. We also plan to improve the model's real time performance by using some adaptive brightness normalization techniques which can help with the inconsistent predictions caused due to poor lighting conditions. Using depth based sensors might help in predictions when there are multiple objects or partial obstructions of the hand, which may confuse the model.

REFERENCES

- [1] Katoch, S., Singh, V. and Tiwary, U.S., 2022. [Indian Sign Language recognition system using SURF with SVM and CNN](https://doi.org/10.1016/j.array.2022.100141). Array, 14, p.100141. or <https://doi.org/10.1016/j.array.2022.100141>
- [2] Viswanathan, Daleesha M., and Sumam Mary Idicula. "[Recent developments in Indian sign language recognition: an analysis](#)." International Journal of Computer Science and Information Technologies 6.1 (2015): 289-293.
- [3] Mitra, Sushmita, and Tinku Acharya. "[Gesture recognition: A survey](#)." IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 37.3 (2007): 311-324.
- [4] Yutong Chen, Fangyun Wei, Xiao Sun, Zhirong Wu, Stephen Lin; Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 2022, pp. 5120-5130. or <https://doi.org/10.48550/arXiv.2203.04287>
- [5] Padmavathi, S., M. S. Saipreethy, and V. Valliammai. "[Indian sign language character recognition using neural networks](#)." IJCA Special Issue on Recent Trends in Pattern Recognition and Image Analysis RTPRIA 1 (2013): 40-45.
- [6] Bhatti, Zeeshan, et al. "[Text to animation for sign language of Urdu and Sindhi](#)." IKSP Journal of Emerging Trends in Basic and Applied Sciences 1.1 (2021).
- [7] Kamruzzaman, M. M. "Arabic sign language recognition and generating Arabic speech using convolutional neural network." Wireless Communications and Mobile Computing 2020.1 (2020): 3685614. or <https://doi.org/10.1155/2020/3685614>
- [8] Vogler, Christian & Metaxas, Dimitris. (2003). Handshapes and Movements: Multiple-Channel American Sign Language Recognition. 247-258. 10.1007/978-3-540-24598-8_23. or https://doi.org/10.1007/978-3-540-24598-8_23
- [9] Das, S., Imtiaz, M. S., Neom, N. H., Siddique, N., & Wang, H. (2023). A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier. Expert Systems with Applications, 213, 118914. <https://doi.org/10.1016/j.eswa.2022.118914>
- [10] Aloysius, N., Geetha, M. Understanding vision-based continuous sign language recognition. Multimed Tools Appl 79, 22177–22209 (2020). <https://doi.org/10.1007/s11042-020-08961-z>
- [11] Sharma, S., Singh, S. Recognition of Indian Sign Language (ISL) Using Deep Learning Model. Wireless Pers Commun 123, 671–692 (2022). <https://doi.org/10.1007/s11277-021-09152-1>

- [12] Chen, Feng-Sheng, Chih-Ming Fu, and Chung-Lin Huang. "[Hand gesture recognition using a real-time tracking method and hidden Markov models](#)." Image and vision computing 21.8 (2003): 745-758.[https://doi.org/10.1016/S0262-8856\(03\)00070-2](https://doi.org/10.1016/S0262-8856(03)00070-2)
- [13] Rahaman, Muhammad Aminur, et al. "Real-time computer vision-based Bengali sign language recognition." 2014 17th international conference on computer and information technology (ICCIT). IEEE, 2014. Or <https://doi.org/10.1109/ICCITech.2014.7073150>
- [14] Awad, George, Junwei Han, and Alistair Sutherland. "A unified system for segmentation and tracking of face and hands in sign language recognition." 18th International Conference on Pattern Recognition (ICPR'06). Vol. 1. IEEE, 2006. or <https://doi.org/10.1109/ICPR.2006.194>
- [15] Johnson, Jane E., and Russell J. Johnson. "[Assessment of regional language varieties in Indian sign language](#)." SIL Electronic Survey Report 6 (2008): 2008.
- [16] Farooq, Uzma, et al. "Advances in machine translation for sign language: approaches, limitations, and challenges." Neural Computing and Applications 33.21 (2021): 14357-14399. Or <http://dx.doi.org/10.1007/s00521-021-06079-3>

APPENDIX-A

PSUEDOCODE

```
# sign_to_speech_translator.py
import cv2
import numpy as np
import tensorflow as tf
import mediapipe as mp
import os
import time

import customtkinter as ctk
from PIL import Image
import tkinter.messagebox as mbox
from gtts import gTTS
from deep_translator import GoogleTranslator
import pygame

# ——— CONFIG —————
ctk.set_appearance_mode("Dark")
ctk.set_default_color_theme("green")
pygame.mixer.init()

MODEL_PATH =
"C:/Users/chakr/OneDrive/Desktop/isl@3/sign_language_mobilenetv2_finetuned_final.keras"
DATASET_DIR = "C:/Users/chakr/OneDrive/Desktop/isl@3/dataset"
model = tf.keras.models.load_model(MODEL_PATH)
labels = sorted(os.listdir(DATASET_DIR))

mp_hands = mp.solutions.hands
mp_drawing = mp.solutions.drawing_utils
hands = mp_hands.Hands(static_image_mode=False, max_num_hands=2,
min_detection_confidence=0.7)
```

```
cap = cv2.VideoCapture(0)
if not cap.isOpened():
    raise RuntimeError("Could not open webcam.")

sign_buffer = []
sentence_buffer = []
capture_start_time = None
recognition_time_threshold = 2
last_prediction = None
last_sign_time = 0
cooldown_time = 1

LANG_OPTIONS = {
    "Hindi": "hi", "French": "fr", "Spanish": "es", "German": "de", "Arabic": "ar", "Chinese
(Simplified)": "zh-CN",
    "Chinese (Traditional)": "zh-TW", "Russian": "ru", "Japanese": "ja", "Korean": "ko",
    "Portuguese": "pt", "Italian": "it",
    "Dutch": "nl", "Turkish": "tr", "Swedish": "sv", "Polish": "pl", "Greek": "el", "Thai": "th",
    "Vietnamese": "vi",
    "Bengali": "bn", "Tamil": "ta", "Telugu": "te", "Urdu": "ur", "Malayalam": "ml", "Punjabi":
    "pa", "Marathi": "mr",
    "Gujarati": "gu", "Malay": "ms", "Indonesian": "id", "Hebrew": "he", "Swahili": "sw",
    "Nepali": "ne", "Sinhala": "si",
    "Kannada": "kn"
}

class SignLangApp(ctk.CTk):
    def __init__(self):
        super().__init__()
        self.title("👋 Sign to Speech Translator")
        self.geometry("1200x700")
        self.grid_columnconfigure((0, 1), weight=1, uniform="col")
```

```
self.grid_rowconfigure(0, weight=1)

def icon(path): return ctk.CTkImage(Image.open(path), size=(20, 20))
speaker_icon = icon("C:/Users/chakr/Downloads/icons8-speaker-24.png")
translate_icon = icon("C:/Users/chakr/Downloads/icons8-translate-24.png")
space_icon = icon("C:/Users/chakr/Downloads/icons8-space-key-24.png")
back_icon = icon("C:/Users/chakr/Downloads/icons8-back-button-24.png")
clear_icon = icon("C:/Users/chakr/Downloads/icons8-delete-history-24.png")

self.video_frame = ctk.CTkFrame(self, corner_radius=16)
self.video_frame.grid(row=0, column=0, padx=20, pady=20, sticky="nsew")
self.video_frame.grid_rowconfigure(0, weight=1)
self.video_frame.grid_columnconfigure(0, weight=1)

self.video_border = ctk.CTkFrame(self.video_frame, corner_radius=10, fg_color="red")
self.video_border.grid(row=0, column=0, padx=4, pady=4, sticky="nsew")

self.video_label = ctk.CTkLabel(self.video_border, text="", fg_color="black")
self.video_label.pack(expand=True, fill="both")

self.ctrl_frame = ctk.CTkFrame(self, corner_radius=16)
self.ctrl_frame.grid(row=0, column=1, padx=20, pady=20, sticky="nsew")
self.ctrl_frame.grid_columnconfigure(0, weight=1)

self.header_label = ctk.CTkLabel(self.ctrl_frame, text="🔊 Sign to Speech Translator",
font=("Arial", 22, "bold"))
self.header_label.grid(row=0, column=0, padx=10, pady=(10, 5), sticky="ew")

self.theme_switch = ctk.CTkSwitch(self.ctrl_frame, text="Dark Mode",
command=self.toggle_theme)
self.theme_switch.select()
self.theme_switch.grid(row=1, column=0, padx=10, pady=(5, 10), sticky="e")
```

```
self.status_label = ctk.CTkLabel(self.ctrl_frame, text="No Hand Detected",
text_color=("red", "red"), anchor="w", font=("Arial", 14))
self.status_label.grid(row=2, column=0, pady=(0, 5), padx=10, sticky="ew")

self.sentence_label = ctk.CTkLabel(self.ctrl_frame, text="Sentence: ", anchor="w",
font=("Arial", 14))
self.sentence_label.grid(row=3, column=0, padx=10, pady=(0, 0), sticky="ew")

self.word_label = ctk.CTkLabel(self.ctrl_frame, text="Word: ", anchor="w",
font=("Arial", 14))
self.word_label.grid(row=4, column=0, padx=10, pady=(5, 0), sticky="ew")

self.sign_label = ctk.CTkLabel(self.ctrl_frame, text="Current Sign: ", anchor="w",
font=("Arial", 14))
self.sign_label.grid(row=5, column=0, padx=10, pady=(5, 10), sticky="ew")

self.translate_box = ctk.CTkFrame(self.ctrl_frame, corner_radius=12)
self.translate_box.grid(row=6, column=0, padx=10, pady=10, sticky="ew")
self.translate_box.grid_columnconfigure(0, weight=1)

ctk.CTkLabel(self.translate_box, text="Translate to:", font=("Arial", 13)).grid(row=0,
column=0, padx=10, pady=(10, 0), sticky="w")

self.lang_menu = ctk.CTkOptionMenu(self.translate_box,
values=list(LANG_OPTIONS.keys()))
self.lang_menu.set("Hindi")
self.lang_menu.grid(row=1, column=0, padx=10, pady=5, sticky="ew")

self.translated_text = ctk.CTkTextbox(self.translate_box, height=80)
self.translated_text.grid(row=2, column=0, padx=10, pady=(5, 10), sticky="ew")

def make_button(text, image, color, command):
    return ctk.CTkButton(self.ctrl_frame, text=text, image=image, fg_color=color,
hover_color="#1a1a1a", corner_radius=12, font=("Arial", 14), command=command)
```



```
        make_button("Speak", speaker_icon, "green", self.speak_sentence).grid(row=7,
column=0, padx=10, pady=5, sticky="ew")
        make_button("Translate", translate_icon, "blue", self.translate_sentence).grid(row=8,
column=0, padx=10, pady=5, sticky="ew")
        make_button("Space", space_icon, "blue", self.add_space).grid(row=9, column=0,
padx=10, pady=5, sticky="ew")
        make_button("Back", back_icon, "orange", self.backspace).grid(row=10, column=0,
padx=10, pady=5, sticky="ew")
        make_button("Clear", clear_icon, "red", self.clear_all).grid(row=11, column=0,
padx=10, pady=5, sticky="ew")
```

```
self.after(10, self.update_frame)
self.bind("<space>", lambda event: self.add_space())
self.bind("<BackSpace>", lambda event: self.backspace())
self.bind("<Escape>", lambda event: self.clear_all())
```

```
def toggle_theme(self):
```

```
    theme = "Dark" if self.theme_switch.get() else "Light"
    ctk.set_appearance_mode(theme)
```

```
def wait_for_stop(self):
```

```
    while pygame.mixer.music.get_busy():
        time.sleep(0.1)
```

```
def play_audio(self, filepath):
```

```
    try:
        pygame.mixer.music.load(filepath)
        pygame.mixer.music.play()
    except Exception as e:
        mbox.showerror("Audio Error", f"Failed to play audio:\n{e}")
```

```
def speak_sentence(self):
```

```
    if not sentence_buffer and not sign_buffer:
        return mbox.showwarning("Speak", "No sentence to speak")
```

```
full = sentence_buffer.copy()
if sign_buffer:
    full.append("".join(sign_buffer))
s = " ".join(full).strip()
if not s:
    return mbox.showwarning("Speak", "No content to speak")
lang = LANG_OPTIONS[self.lang_menu.get()]
pygame.mixer.music.stop()
self.wait_for_stop()
filename = "output.mp3"
if os.path.exists(filename):
    os.remove(filename)
tts = gTTS(text=s, lang=lang, slow=False)
tts.save(filename)
self.play_audio(filename)

def translate_sentence(self):
    full = sentence_buffer.copy()
    if sign_buffer:
        full.append("".join(sign_buffer))
    s = " ".join(full).strip()
    if not s:
        return mbox.showwarning("Translate", "Nothing to translate")

    tgt = LANG_OPTIONS[self.lang_menu.get()]
    try:
        out = GoogleTranslator(source="auto", target=tgt).translate(s)
        self.translated_text.delete("0.0", "end")
        self.translated_text.insert("0.0", out)

        pygame.mixer.music.stop()
        self.wait_for_stop()
        filename = "translated_output.mp3"
        if os.path.exists(filename):
```

```
        os.remove(filename)

        tts = gTTS(text=out, lang=tgt, slow=False)
        tts.save(filename)
        self.play_audio(filename)

    except Exception as e:
        mbox.showerror("Translate Error", str(e))

    def add_space(self):
        if sign_buffer:
            sentence_buffer.append("".join(sign_buffer))
            sign_buffer.clear()
            self.update_labels()

    def backspace(self):
        if sign_buffer:
            sign_buffer.pop()
            self.update_labels()

    def clear_all(self):
        sign_buffer.clear()
        sentence_buffer.clear()
        self.translated_text.delete("0.0", "end")
        self.update_labels()

    def update_labels(self):
        self.word_label.configure(text=f"Word: { ''.join(sign_buffer) }")
        self.sentence_label.configure(text=f"Sentence: { ' '.join(sentence_buffer) }")
        self.sign_label.configure(text=f"Current Sign: { sign_buffer[-1] if sign_buffer else ''}")

    def update_frame(self):
        global capture_start_time, last_prediction, last_sign_time
        ret, frame = cap.read()
        if not ret:
```

```
    return
    img_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)
    results = hands.process(img_rgb)
    border_color = "red"
    if results.multi_hand_landmarks:
        border_color = "green"
        self.status_label.configure(text="Hand Detected", text_color=("green", "green"))
        h, w, _ = frame.shape
        xs, ys = [], []
        for handLms in results.multi_hand_landmarks:
            for lm in handLms.landmark:
                xs.append(int(lm.x * w))
                ys.append(int(lm.y * h))
            mp_drawing.draw_landmarks(frame, handLms,
mp_hands.HAND_CONNECTIONS)
        if xs and ys:
            x1, x2 = max(min(xs) - 40, 0), min(max(xs) + 40, w)
            y1, y2 = max(min(ys) - 40, 0), min(max(ys) + 40, h)
            roi = frame[y1:y2, x1:x2]
            if roi.size:
                roi = cv2.resize(cv2.cvtColor(roi, cv2.COLOR_BGR2RGB), (160, 160)) / 255.0
                pred = model.predict(np.expand_dims(roi, 0), verbose=0)
                lbl = labels[np.argmax(pred)]
                now = time.time()
                if lbl == last_prediction:
                    if capture_start_time is None:
                        capture_start_time = now
                    elif now - capture_start_time >= recognition_time_threshold:
                        if now - last_sign_time > cooldown_time:
                            sign_buffer.append(lbl)
                            last_sign_time = now
                            capture_start_time = None
                else:
                    capture_start_time = None
```

```
        last_prediction = lbl
        self.update_labels()
    else:
        self.status_label.configure(text="No Hand Detected", text_color=("red", "red"))
        capture_start_time = None

    img = Image.fromarray(img_rgb)
    ctk_img = ctk.CTkImage(img, size=(640, 480))
    self.video_label.configure(image=ctk_img)
    self.video_label.image = ctk_img
    self.video_border.configure(fg_color=border_color)
    self.after(10, self.update_frame)

if __name__ == "__main__":
    app = SignLangApp()
    app.mainloop()
    cap.release()
```

APPENDIX-B

SCREENSHOTS

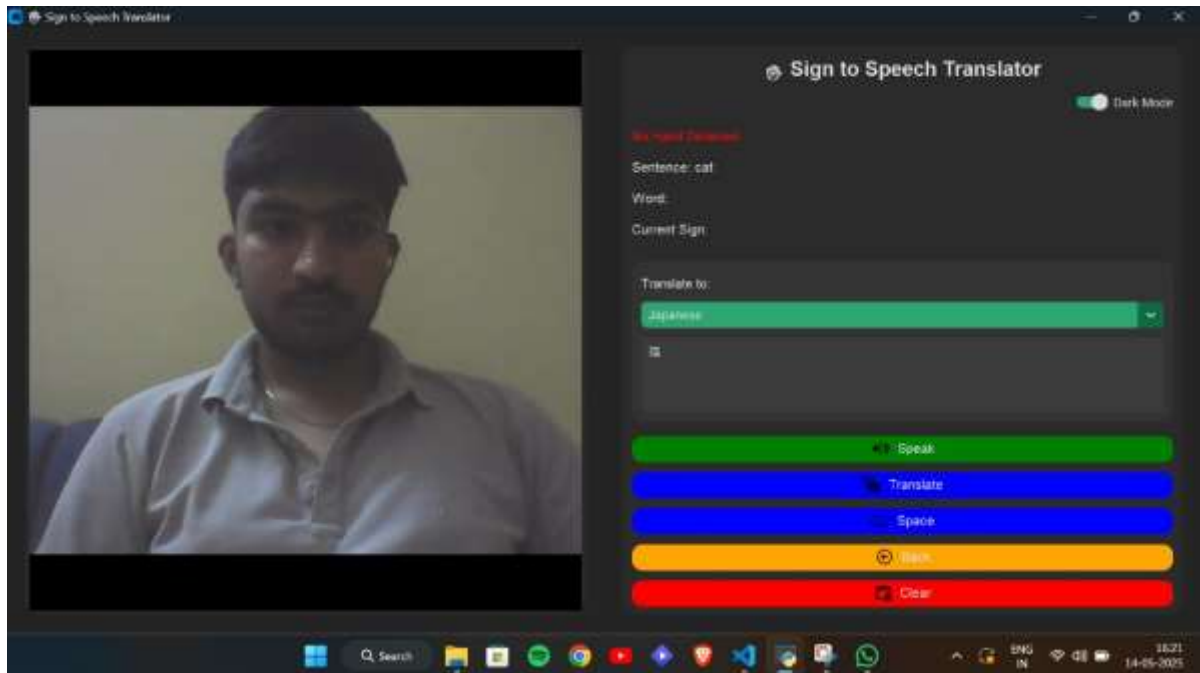


Fig 11. 1 Input from detected signs

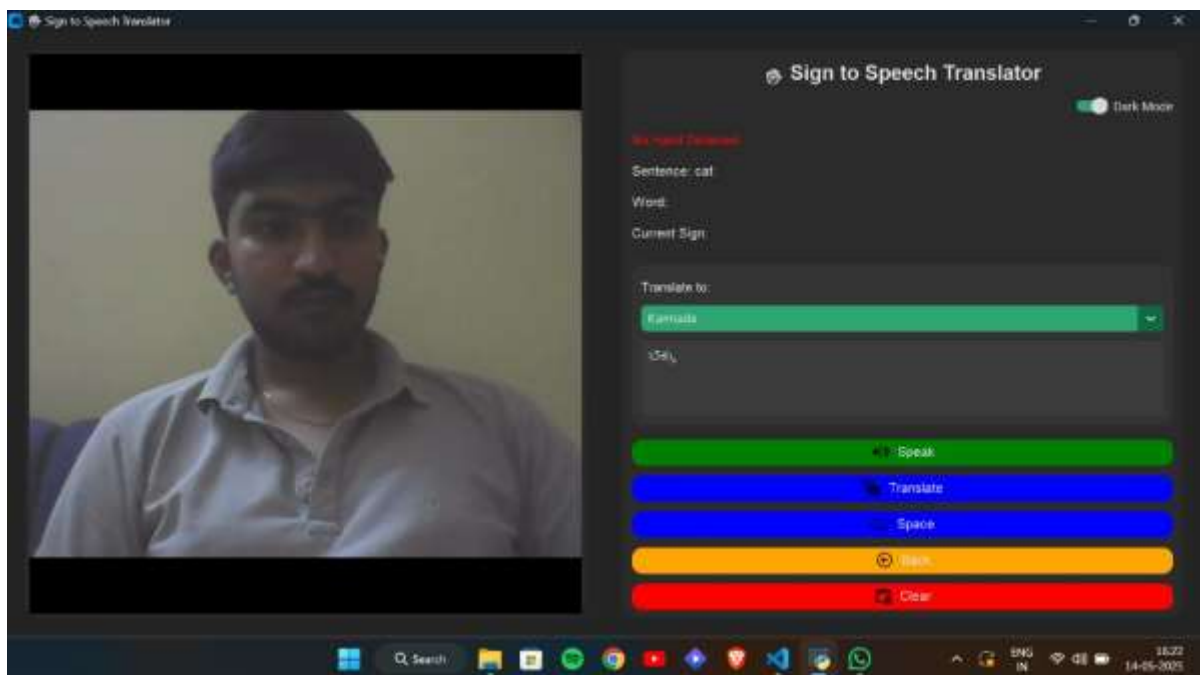


Fig 11. 2 Input translated to kannada

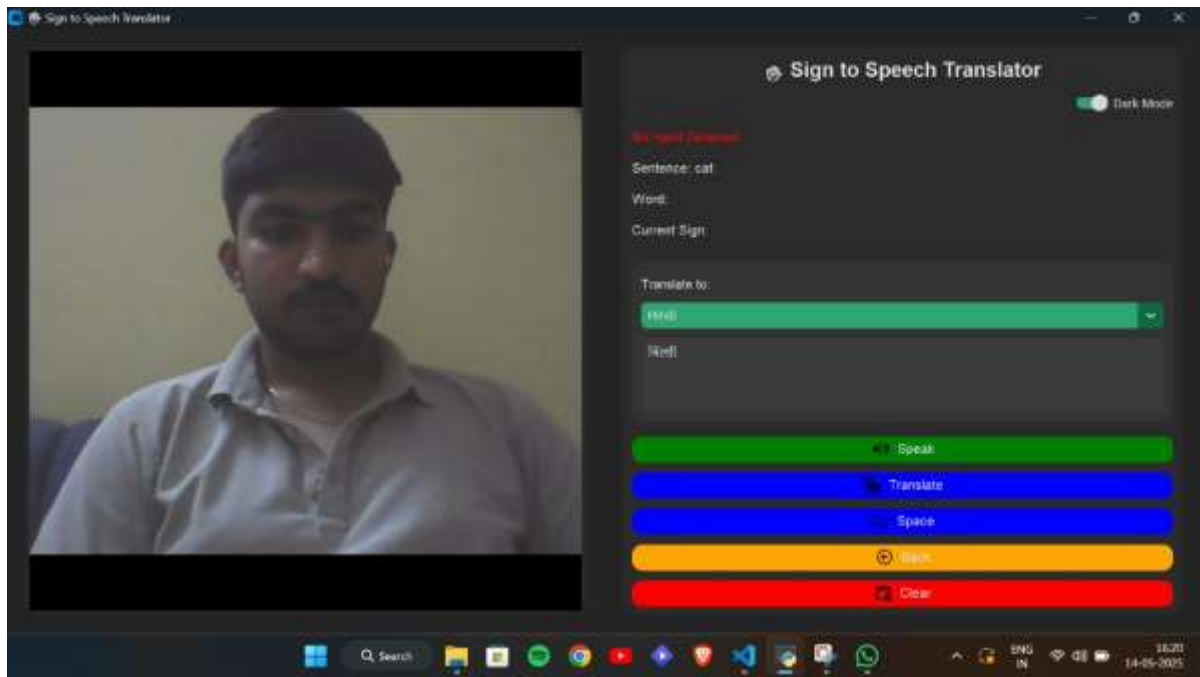


Fig 11. 3 Input translated into Hindi

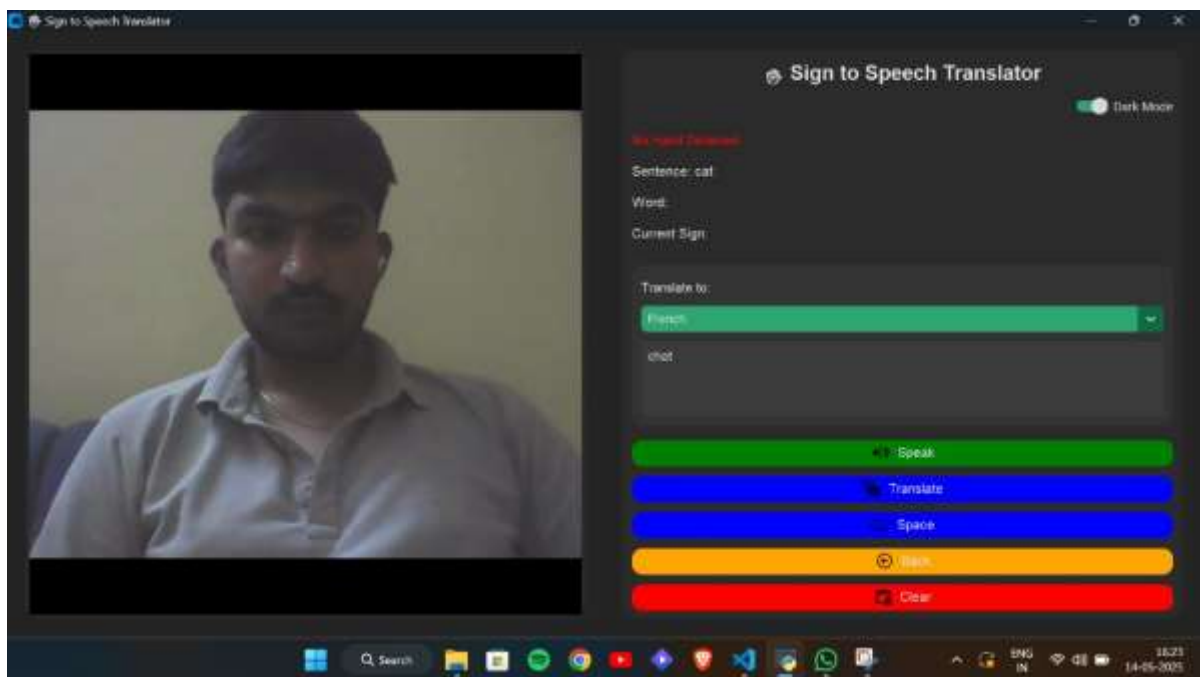


Fig 11. 4 Input translated into French

APPENDIX-C

ENCLOSURES

Paper Submission (ICCAMS2025):

The screenshot shows the 'Author Console' for ICCAMS2025. At the top, there are navigation links: Submissions, Contact Chairs, Help Center, Select Your Role, Author, ICCAMS2025, and Hasan Raza. Below the navigation bar, there's a section for 'Author Console' with a link to 'Please click here to view Welcome Message & Instructions.' A table lists the submitted papers. The first paper is 'Indian Sign Language to Text/Speech translation' with ID 918. The abstract is visible, and the submission files are listed as 'CAI_G26_Updated_draft_paper.docx' and 'Supplementary File Not Uploaded'.

Plagiarism Test/Similarity Index:

The screenshot shows a Turnitin plagiarism report. The top bar includes the Turnitin logo, 'Page 2 of 36 - Integrity Overview', and the Submission ID 'trnoid::1:3249644834'. The main heading is '17% Overall Similarity', with a note that this is the combined total of all matches, including overlapping sources. Below this, there's a section for 'Filtered from the Report' with a dropdown menu set to 'Bibliography'. The 'Match Groups' section shows four categories: '93 Not Cited or Quoted 17%' (Matches with neither in-text citation nor quotation marks), '0 Missing Quotations 0%' (Matches that are still very similar to source material), '0 Missing Citation 0%' (Matches that have quotation marks, but no in-text citation), and '0 Cited and Quoted 0%' (Matches with in-text citation present, but no quotation marks). The 'Top Sources' section lists three sources: 'Internet sources' (3%), 'Publications' (13%), and 'Submitted works (Student Papers)' (11%). The 'Integrity Flags' section shows '0 Integrity Flags for Review' and a note that 'No suspicious text manipulations found.' A blue box at the bottom contains a message from the system's algorithms, stating that they look deeply at a document for any inconsistencies and that a flag is not necessarily an indicator of a problem, but a recommendation for further review.

SUSTAINABLE DEVELOPMENT GOALS



- **SDG 3: Good Health and Well-being:** Improved communication through sign language helps in healthcare settings, ensuring that patients with hearing impairments receive proper information, care, and emotional support.
- **SDG 4: Quality Education:** Project promotes inclusive education by helping individuals with hearing and speech impairments access learning materials through sign language, breaking communication barriers. Also eliminate gender disparities and ensure equal access to all levels of education and vocational training for the vulnerable, including persons with disabilities.
- **SDG 9: Industry, Innovation, and Infrastructure:** Using technology for sign language translation represents innovation. It contributes to building accessible communication infrastructure and inclusive tech solutions.
- **SDG 10: Reduced Inequalities:** By translating spoken or written content into Indian Sign Language, your project empowers the deaf and hard-of-hearing community, promoting social inclusion and reducing inequality in communication and access to services.

- **SDG 16: Peace, Justice and Strong Institutions:** Promoting accessible communication aligns with inclusive institutions and ensures equal access to justice and public services for all, including persons with disabilities.