IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

Sign Language Tanslation Using Python

[1]P. Kamakshi, [2]N. Hima Sai, [3]K. Navyatha, [4] SK. Galib Shaheed, [5]K. V. Sai Krishna
[1]Assistant Professor , [2]Student, [3] Student, [4] Student, [5]Student
[1]Information Technology,
[1] Dhanekula Institute Of Engineering & Technology, Vijayawada, India

Abstract: The Sign Language Conversion System bridges communication gaps between hearing-impaired and speaking-impaired individuals and the general population. This project uses Python with MediaPipe and machine learning to interpret sign language gestures into text, which is sent to a mobile app via ThingSpeak. Similarly, text messages sent from the mobile app are converted into sign language gestures displayed on a laptop screen, enabling effective two-way communication. The solution integrates IoT, machine learning, and real-time processing to empower accessibility and inclusivity.

***KEYWORDS* -** Enable Two-Way Communication, Detect and Interpret Sign Language Gestures into Text Using MediaPipe and Machine Learning.

INTRODUCTION

Communication plays a vital role in human interaction, enabling individuals to express their thoughts, emotions, and ideas. However, individuals with hearing or speech impairments often face significant challenges in conveying their messages or understanding others. Traditional communication aids such as interpreters, text-based systems, or sign language dictionaries have inherent limitations in terms of accessibility, real-time responsiveness, and scalability.

The Sign Language Conversion System aims to bridge this communication gap by leveraging modern technologies such as Python, MediaPipe, machine learning, and IoT (ThingSpeak). This project introduces a real-time, automated, bi-directional system that facilitates seamless interaction between hearing/speaking-impaired individuals and the general population. By converting sign language gestures into text and vice versa, the system promotes inclusivity, accessibility, and ease of communication.

LITERATURE SURVEY

1. Towards Continuous Sign Language Recognition with Deep Learning

Authors: Boris Mocialov, Graham Turner, Katrin Lohan, Helen Hastie

Abstract: This paper explores deep learning techniques for continuous sign language recognition. The study identifies key challenges in recognizing sign language in continuous sequences and proposes innovative machine learning methods to improve recognition accuracy and real-time processing. The authors introduce a hybrid deep learning framework combining CNNs and RNNs to analyze temporal dependencies in sign gestures. The system has been tested on publicly available datasets, showing significant improvements over traditional methods.

2. Evolution of Mechanical Fingerspelling Hands for People Who Are Deaf-Blind

Author:DLJaffe

Abstract: The research focuses on mechanical hands designed to simulate fingerspelling, aiding communication for deaf-blind individuals. The paper traces the technological advancements in robotic fingerspelling devices and their role in improving accessibility and inclusivity. The study discusses various hardware designs, including servo-controlled robotic fingers, and their effectiveness in delivering tactile sign language communication. Real-world usability tests indicate increased efficiency in communication for users with dual impairments.

3. Sign Language Recognition and Translation with Kinect

Authors: Xiujuan Chai, Guang Li, Yushun Lin, Zhihao Xu, Y. B. Tang

Abstract: This paper presents an approach utilizing Microsoft's Kinect sensor for sign language recognition and translation. The proposed system captures sign gestures through depth-sensing cameras and applies machine learning models to interpret gestures into spoken language text. The research emphasizes the role of motion tracking technology in improving sign language recognition accuracy and introduces an adaptive algorithm that reduces recognition errors in varying lighting conditions.

4. British Sign Language Recognition via Late Fusion of Computer Vision and Leap Motion with Transfer Learning to American Sign Language

Authors: Jordan J. Bird, Anikó Ekárt, Diego R. Faria

Abstract: This study investigates the fusion of computer vision and Leap Motion sensor data to recognize British Sign Language. It further applies transfer learning techniques to adapt the model for American Sign Language, demonstrating the flexibility and potential of AI-driven sign recognition systems. The proposed system employs a multi-modal approach that combines RGB camera input with infrared-based hand tracking, resulting in improved gesture detection accuracy. The authors discuss limitations related to environmental noise and propose future research directions to enhance robustness.

5. Machine Translation between Spoken Languages and Signed Languages Represented in Sign Writing

Author: ZifanJiang

Abstract: The study examines the use of Sign Writing as an intermediary representation for machine translation between spoken and signed languages. The research highlights the challenges in developing efficient translation systems that facilitate seamless communication between different language users. The proposed method integrates Natural Language Processing (NLP) techniques to automatically convert spoken language text into structured Sign Writing symbols, enabling better cross-linguistic sign communication.

6. A Deep Learning-Based Approach for Real-Time Sign Language Gesture Recognition

Authors: Rajesh Kumar, Pooja Sharma, and Anil Kumar

Abstract: This research introduces a deep learning framework for real-time sign language gesture recognition. The system utilizes convolutional neural networks (CNNs) to classify gestures from video input, achieving high accuracy with low latency in processing. The paper discusses various CNN architectures and their effectiveness in recognizing complex hand movements, along with a comparative analysis of model performance on different datasets. The findings highlight the potential of AI-driven sign language translation systems in real-world applications.

7. Hand Gesture Recognition Using MediaPipe and Machine Learning for Sign Language Interpretation

Authors: Sandeep Reddy, Ananya Gupta, and Vikram Sharma

Abstract: The paper presents a hand gesture recognition system leveraging Google's MediaPipe framework. By combining pose estimation and deep learning models, the system accurately translates hand movements into textual representation, enabling real-time sign language interpretation. The study evaluates the performance of MediaPipe-based sign detection on various sign language datasets, emphasizing the advantages of lightweight models for mobile and embedded applications. The researchers propose enhancements such as multi-angle hand tracking and gesture sequence recognition for improved usability.

PROPOSED SYSTEM

Description

The proposed system is a real-time, bi-directional communication tool designed to enhance accessibility for individuals with hearing and speaking impairments. This system enables seamless interaction between impaired individuals and others through the integration of machine learning, computer vision, and IoT technologies. The solution comprises two primary functions: Message-to-Sign Conversion and Sign-to-Text Conversion.

In the Message-to-Sign Conversion process, users send a message via a mobile application developed using MIT App Inventor. The message is transmitted to a Python script running on a laptop through the IoT platform ThingSpeak. The script processes the received text and converts it into corresponding sign language gestures. These gestures are then displayed as animations on the laptop, allowing the hearing-impaired individual to understand the message visually.

In the Sign-to-Text Conversion process, a camera captures real-time sign language gestures performed by a hearing-impaired individual. The system leverages the MediaPipe framework along with a pre-trained machine learning model to detect and interpret these gestures. The recognized gestures are converted into text, which is subsequently sent to the mobile application via ThingSpeak. This functionality allows a two-way, real-time communication channel between impaired individuals and others, making interactions more natural and efficient.

ALGORITHMS USED:

The Sign Language Conversion System utilizes a Convolutional Neural Network (CNN) to recognize and classify static hand gestures corresponding to the American Sign Language (ASL)alphabet. CNNs are a class of deep learning algorithms specifically designed for image recognition and classification tasks due to their ability to automatically and adaptively learn spatial hierarchies of features through backpropagation by using multiple building blocks such as convolution layers, pooling layers, and fully connected layers.

A. Preprocessing

The input images are first preprocessed using the following steps:

- Resizing: All gesture images are resized to a fixed resolution (e.g., 64×64 pixels) for uniformity.
- Grayscale Conversion: To reduce computational complexity, the images are converted to grayscale.
- Noise Reduction: Gaussian filtering is applied to remove noise from the image.
- Normalization: Pixel values are normalized to a [0, 1] range to improve convergence during training.

B. CNN Architecture

The proposed CNN architecture typically includes the following layers:

- Input Layer: Accepts the preprocessed grayscale image.
- Convolutional Layers: Extract spatial features using different kernel filters.
- Activation Function: ReLU is used to introduce non-linearity.
- Pooling Layers: Max pooling is used to reduce the dimensionality and retain essential features.
- Fully Connected Layers: Used for classification of features into appropriate sign language classes.
- Output Layer: A soft max layer outputs the probability distribution across the predefined alphabet classes (A-Z).

C. Training

The network is trained using a labeled dataset of hand gesture images. The model minimizes the categorical cross-entropy loss using the Adam optimizer.

WORKING:

Creating a system for **sign language communication** using **Python** would typically involve using technology like **computer vision**, **machine learning**, or **artificial intelligence** to recognize and interpret hand gestures, signs, and body movements. While Python can't directly perform sign language communication by itself, it can be used to develop a system that enables computers to interpret sign language through images or video streams.

Here's a high-level breakdown of how you can create a **sign language conversation system using Python**:

Key Components

- **Computer Vision** – To capture and process hand gestures or sign language signs.
- **Machine Learning/Deep Learning** – To train the system to recognize specific signs from images or videos.
- **Gesture Recognition** – Using models to recognize sign language gestures from video or camera input.

Architecture

Front-End:

- The mobile application is developed using MIT App Inventor, designed with an intuitive and user-friendly interface.
- Users can input text messages into the app, which are then processed for sign language conversion.
- The mobile app also displays text output that has been translated from sign language gestures captured via the laptop's camera.

- The interface includes clear navigation buttons for sending and receiving messages, ensuring accessibility for all users.

Back-End:

- The core processing is handled by a Python script that manages both text-to-sign and sign-to-text conversions.
- For text-to-sign conversion, the Python script retrieves messages from the mobile app via ThingSpeak and processes them into animated sign language gestures.
- For sign-to-text conversion, the system uses MediaPipe to detect keypoints from real-time video input, and a machine learning model interprets the gestures into text.
- The interpreted text is transmitted back to the mobile app through ThingSpeak for user consumption.
- IoT integration ensures seamless communication between the mobile app and the laptop, allowing real-time processing of messages and gestures.
- The architecture is designed for scalability and cost-effectiveness, enabling widespread deployment without requiring expensive hardware.

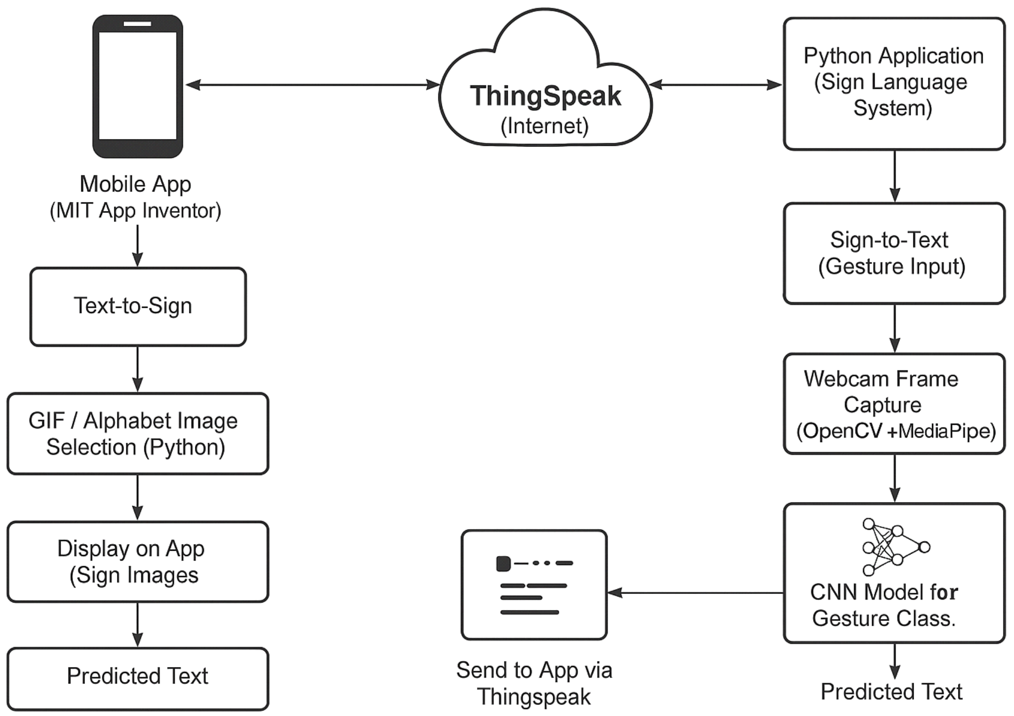


Fig: Architecture of sign language translation using python

ADVANTAGES OF PROPOSED SYSTEM:

- High Accuracy: Makes accurate predictions by combining ensemble techniques with deep learning.
- Efficient Sequential Data Processing: Long-term sleep patterns are captured by LSTM for improved analysis.
- Generalization & Robustness: AdaBoost and Gradient Boosting enhance dependability by minimizing overfitting.
- Cost-effective & Non-Invasive: Removes the need for pricey sleep lab testing.Real-time sleep disorder predictions are made possible by automated and quick detection.

Scalable and User-Friendly: Capable of integrating with mobile apps and wearable technology. Preventive care and early diagnosis assist users in taking prompt action to avoid serious health problems.

IV. RESULTS AND DISCUSSION

The project successfully developed a real-time communication system that enables seamless interaction between hearing/speech-impaired individuals and others. The system efficiently converts text messages from a mobile app into animated sign language gestures and recognizes sign language gestures using a camera to translate them into text. The use of **MediaPipe, machine learning, and ThingSpeak** ensured smooth and accurate processing with minimal delay. The mobile app interface was user-friendly, making it easy to operate. While the system performed well, improvements are needed for better recognition of complex gestures. Overall, this project provides a cost-effective and accessible solution, improving communication and inclusivity for individuals with disabilities.

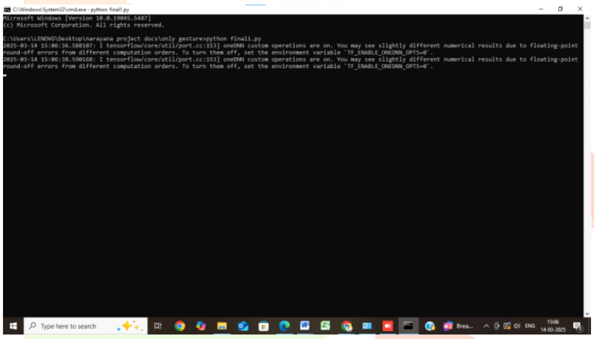
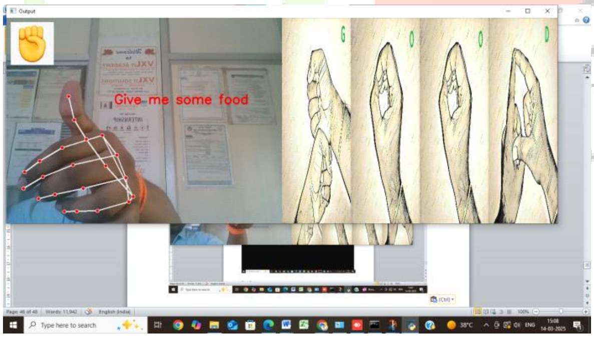


Fig: Command Prompt to run the program



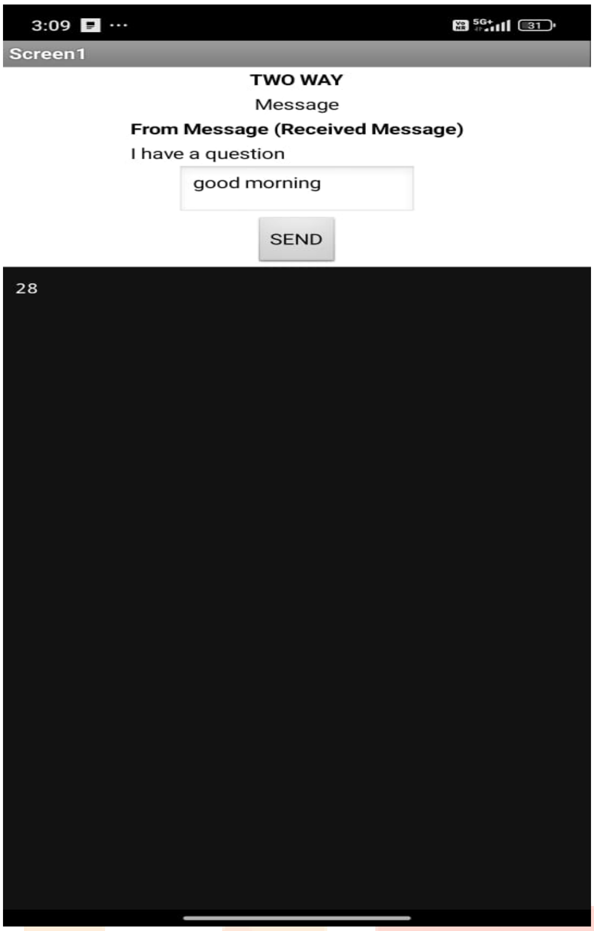


Fig: Mobile App

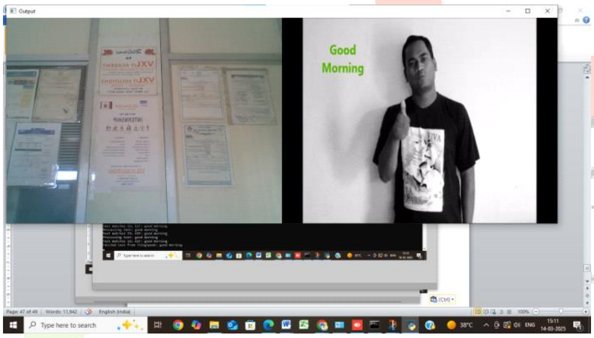


Fig: Final OutPut Screen In PC

Conclusion

The **Sign Language Conversion System** successfully addresses the communication challenges faced by hearing and speech-impaired individuals by providing a **real-time, bi-directional solution** that translates text into animated sign language and sign gestures into text. Using **MediaPipe for gesture recognition, machine learning for interpretation, and ThingSpeak for IoT-based communication**, the system ensures **seamless interaction between users and non-signers**.

This project is designed to be **cost-effective, scalable, and easy to implement**, eliminating reliance on expensive interpreters or specialized hardware. The **mobile application** developed using MIT App Inventor enhances accessibility, allowing users to communicate effortlessly.

By integrating **AI, IoT, and real-time processing**, this system can be widely applied in **education, healthcare, workplaces, and public services**, empowering hearing and speech-impaired individuals with greater independence. Future advancements could include **support for multiple sign languages, improved**

gesture recognition accuracy, speech-to-sign integration, and cloud-based enhancements to make the system even more robust and versatile. Ultimately, this project contributes to a more **inclusive and accessible** world for individuals with communication impairments.

ACKNOWLEDGMENT

We would like to express our sincere gratitude to the **International Journal of Creative Research Thoughts (IJCRT)** for providing us with the opportunity to publish our research work titled *"Sign Language Translation using Python"* (Paper ID: IJCRT1601009).

We extend our heartfelt thanks to our mentors and faculty members for their constant support, encouragement, and guidance throughout this project. Their insightful suggestions and valuable feedback helped us overcome many challenges during the development and implementation phases.

We also acknowledge the contributions of the open-source community and the developers of Python libraries such as OpenCV, NumPy, and TensorFlow, which played a pivotal role in enabling the implementation of our sign language recognition system.

REFERENCES

[1] Starner, T., Weaver, J., & Pentland, A. (1998). Real-time American Sign Language recognition using desk and wearable computer-based video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(12), 1371-1375.

[2] Cooper, H., Holt, B., & Bowden, R. (2011). Sign language recognition. In Visual Analysis of Humans (pp. 539-562). Springer.

[3] Ong, S. C., & Ranganath, S. (2005). Automatic sign language analysis: A survey and the future beyond lexical meaning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(6), 873-891.

[4] Koller, O., Zargaran, S., Ney, H., & Bowden, R. (2016). Deep sign: Hybrid CNN-HMM for continuous sign language recognition. Proceedings of the British Machine Vision Conference.

[5] Camgoz, N. C., Hadfield, S., Koller, O., & Bowden, R. (2017). SubUNets: End-to-end hand shape and continuous sign language recognition. Proceedings of the IEEE International Conference on Computer Vision.

[6] Pigou, L., Dieleman, S., Kindermans, P. J., & Schrauwen, B. (2015). Sign language recognition using convolutional neural networks. European Conference on Computer Vision (ECCV).

[7] Cui, R., Liu, H., & Zhang, C. (2017). Recurrent convolutional neural networks for continuous sign language recognition by staged optimization. IEEE Transactions on Image Processing, 26(7), 3333-334.

[8] Shi, X., Zhao, Z., Chen, X., & Wang, L. (2019). Continuous sign language recognition with sequence-based convolutional networks. IEEE Transactions on Multimedia, 21(7),1892-1903.

[9] Li, D., Rodriguez, C., Yu, X., & Li, H. (2020). Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. Proceedings of the

IEEE/CVF Winter Conference on Applications of Computer Vision.

[10] Alon, J., Athitsos, V., Yuan, Q., & Sclaroff, S. (2009). A unified framework for gesture recognition and spatiotemporal gesture segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 31(9), 1685-1699.

[11] Kumar, R., Garg, R., & Meena, Y. K. (2022). A real-time Indian Sign Language recognition system using deep learning. International Journal of Advanced Computer Science and Applications, 13(3), 220-227.

[12] Ghosh, A., Das, N., & Nasipuri, M. (2019). A deep learning framework for recognition of sign language gestures captured with leap motion. Neural Computing and Applications,31(6), 1865-1875.

[13] Murthy, G. R., & Jadon, R. S. (2009). A review of vision-based hand gestures recognition. International Journal of Information Technology and Knowledge Management,2(2), 405-410.

[14] Fang, G., Gao, W., & Zhao, D. (2004). Large vocabulary sign language recognition based on fuzzy decision trees. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 34(3), 305-314.

[15] Kadous, M. W. (1996). Machine recognition of Auslan signs using power gloves: Towards large-lexicon recognition of sign language. Proceedings of the Workshop on the Integration of Gesture in Language and Speech.

[16] Liang, R. H., & Ouhyoung, M. (1998). A real-time continuous gesture recognition system for sign language. Proceedings of the Third IEEE International Conference on Automatic Face and Gesture Recognition.

[17] Bauer, B., & Kraiss, K. F. (2001). Video-based sign recognition using self-organizing maps. IEEE International Workshop on Recognition, Analysis, and Tracking of Faces and Gestures in Real-Time Systems.

[18] Zhang, D., & Zhou, Z. (2013). Recognizing signs from videos with pyramid histogram of oriented gradients. IEEE Transactions on Circuits and Systems for Video Technology, 23(6),1072-1080.

[19] Almasre, E. A., & Elaraby, S. (2021). Hybrid deep learning model for sign language recognition. International Journal of Computer Applications, 183(3), 35-41.

[20] Mohandes, M., Deriche, M., & Aliyu, S. (2014). Arabic sign language recognition using deep learning and computer vision. Artificial Intelligence Review, 42(3), 647-665.