ELSEVIER

Scopus

doi

Journal of Vibration Engineering

ISSN:1004-4523

Registered

- SCOPUS
- GOOGLE SCHOLAR
- DIGITAL OBJECT IDENTIFIER (DOI)
- IMPACT FACTOR 6.1

Our Website
www.jove.science

Real-Time Text to Sign language conversion using deep learning with 3d avatar

Yetukuri Gana Bala Meghana[1]
Student
Department of Computer Science and Engineering
Karunya Institute of Technology and Sciences
India

Beereddy Sreeja[2]
Student
Department of Computer Science and Engineering
Karunya Institute of Technology and Sciences
India

Mokara Anjali[3]
Student
Department of Computer Science and Engineering
Karunya Institute of Technology and Sciences
India

Dr. R. Chitra[4]
Associate Professor
Department of Computer Science and Engineering
Karunya Institute of Technology and Sciences
India

***Abstract*—While the entire community of patients suffering from impaired hearing and speech greatly relies on sign language, inclusive communication with nonsigners is yet impeded due to a lack of effective translation services. This work proposes an avatar-based toolkit, Sign-Kit, for real-time Indian Sign Language (ISL) gesture recognition and visualization. The system integrates a module based on deep learning-based picture categorization using a curator-trained dataset of static hand gestures representing the set of ISL alphabets. The proposed system consists of two modules: a progressive web application interface and a light-weight convolutional neural network (CNN), which provides highly accurate complicated gesture detection and 3D avatar visualization for translation of gesture. A scalable back-end system with Node.js and MongoDB ensures that data processing and user administration are efficiently handled. The front-end interface has been designed user-friendly for both learners and interpreters. Experimental results show that the Sign-Kit provides consistent accuracy in gesture detection, hence providing a practical and easy-to-use system that can assist in breaking down the deterrents in communication between signers and nonsigners.**

***Index Terms*—Sign Language Recognition, Indian Sign Language (ISL), Deep Learning, Convolutional Neural Network (CNN), Avatar-based Translation, Human–Computer Interaction, Accessibility Technology.**

I. Introduction

Millions of deaf and hard-of-hearing persons are regularly denied the opportunity to communicate with the hearing population, though communication is a basic human need. Over time, sign language has evolved as a natural and expressive way in which the Deaf people can communicate through body language, hand gestures, and facial emotions. Nonetheless, there exists a gap in communication due to non-signers' ignorance and lack of a universal interpreter. There lies a great demand for technology-driven assistive solutions since, though Indian Sign Language, or ISL, is the most used sign language in India, it is mainly rendered by manual interpretation. The recent rise of computer vision and deep learning allows making significant advances in gesture detection systems. Furthermore, determining both static and dynamic hand motions using CNN and real-time picture processing approaches has also been successfully completed by researchers. However, most of the existing approaches suffer from computational complexity, dependence on diverse datasets and technology, and a lack of user-friendly visualization tools. Sign-Kit's three major goals are: to get an accurate and lightweight model for ISL alphabet static gesture detection. The goal is to create a user avatar-based real-time visualization system for sign-to-animation and text/sigmatron translation systems. Accessibility should be supported with an application that is user-friendly, browser-based, and suitable for educational and communicative purposes. It would promote social inclusion and digital accessibility by enhancing communication between the general public and those with hearing impairments. Sign-Kit sets out to show how deep learning, web technologies, and avatar animation are combined to create a scalable and intuitive framework that leverages real-time ISL recognition and translation.

II. Literature Survey

Prabhakar et al. [1] proposed a sign language recognition system, which converted sign language into text and speech, improving the communication system amongst speech and hearing impaired persons. They used image processing methods to recognize the hand motion to translate into written output. This approach worked well for static and gestures but was unrealistic for dynamic or continuous signs. Generalization to interactive communication systems was also not possible because of its real-time nature and display.

Grover et al. [2] presented a detailed analysis of sign language translation systems for people with hearing and speech impairments. After the sources revealed a deficit in model generalization, dataset availability, and recognition for other sign languages, Axadbegi and co-authors evaluated different computer vision and sensor-based techniques. In addition, they concluded that future research should focus on multimodal integration, real-time translation, and providing an easy interface for the users in order to make these systems more accessible.

An assistive glove-based system, translating Arabic Sign Language (ArSL) gestures into voice, was proposed by Alzubaidi et al. [3]. Hand movements were tracked through flex and motion sensors included in the glove and were translated into digital signals that operated a machine learning model. This sensing-based system, however, brought only an improvement in real-time recognition but remains inappropriate for large-scale deployment or cost-sensitive applications because of its dependency on hardware.

A framework for translating speech to ISL, for two-way communication between hearing and non-hearing individuals, was proposed by Monga et al. [4]. For example, their model conveyed what was being said to students by translating gesture language into tangible, animated models. However, because the system was more about the conversion of text signals to sign language rather than interpretation from visual signals of gestures, it was also an innovative idea for fusing the modalities in audio-visual conversions.

Sharma et al. [5] proposed an NLP-based translation system that could translate spoken and written words into ISL gestures. The model utilized the linguistic structure-the graph- and semantic linkages to generate grammatically valid ISL sequences. Their study demonstrated that while NLP can be used to translate sign language, complete automation requires the use of vision-based recognition systems in addition to NLP.

Peguda et al. [6] present a sign language translation system, which can handle several Indian languages, taking speech as input. ISL representations were generated by incorporating text normalization and speech recognition modules, as well as gesture rendering modules. The reverse translation from sign to text or voice is constrained because the study had scalability as well as multilinguality assessed, and no visual gesture recognition front-end module was considered.

The inclusion of signed languages in mainstream Natural Language Processing (NLP) studies is a necessity, as mentioned in Yin et al. [7](2021). The paper was presented at ACL 2021 and highlighted the linguistic diversity of signed languages and some of the issues that need to be addressed, such as tokenization, multimodal representations, and the lack of real-world datasets. The paper also promoted community-driven approaches for data collection and linguistics-based modeling. Nevertheless, unlike CNN-based categorization frameworks and real-time visual gestures, it mainly focuses on NLP aspects.

Kahlon and Singh [8] presented a critical review of text-to-sign language translation for state-of-the-art models based on motion synthesis and deep learning. Different rule-based, statistical, and neural model techniques were classified together with their advantages and disadvantages. This way, they found a persistent gap between the meaning of text and sign language and concluded that hybrid models of linguistic and visual conceptualizing were necessary to map semantic profiles.

Ajay et al. [9] employed a random forest classifier trained on the gesture dataset in picture form to present an Indian Sign Language recognition model. In conclusion, EqAch was found to give good accuracy with a relatively low computing load, proving that the standard machine learning approach can work for static gesture classification. The model did not yield good performance on a large data set and real-time recognition, though.

Alvin et al.[10] (2021) proposed a K-Nearest Neighbor classifier-based MediaPipe-based hand gesture detection system using KNN classifier for American Sign Language (ASL). The proposed framework utilized Google's MediaPipe for hand landmark coordinate collection and a lightweight machine learning classifier for motion classification. However, the feature-based approach and traditional classification approach were not suitable for scalability and stability under varying environmental conditions, even though it showed computational efficiency and real-time capability using RGB cameras. Additionally, there were no implementations of deep learning-based spatial feature extraction and avatar-based visualization.

A layer-based structure for real-time sign recognition that takes into account hand gesture and movement recognition was proposed by Ibarguren et al.[11] (2010). The precise classification of hand gestures in real-time situations was enabled by the use of accelerometers and data gloves to capture movement information. Although the hardware-based approach guaranteed precise performance, its dependency on wearable devices impeded its applicability and scalability in real-world communication situations. In contrast, vision-based deep learning models make sign language recognition systems more affordable and accessible by eliminating the need for special devices.

Because of its applications in robotics, virtual reality, human–computer interface (HCI), sign language interpretation, and smart surveillance systems, hand gesture recognition (HGR) has become an important area of study in computer vision. A thorough analysis of methods utilized in vision-based hand gesture recognition systems can be found in the work by Oudah et al.[12] (2020).

Stoll et al. [13] proposed a new text-to-sign language translation system that combined the use of 3D virtual avatars with neural machine translation. This technology provided a more engaging communication bridge for the deaf community by translating spoken or written text into sign sequences and displaying the same with the use of a realistic animated avatar. Despite the fact that this system made it easier to translate long text-based information, it still struggled with challenges in the areas of naturalistic gesture production, body articulation accuracy, and expressiveness of the signer.

Hrishikesh et al.[14] (2024) presented a comprehensive vision-based gesture recognition framework aimed at im-

proving natural human–computer interaction through image-based deep learning techniques. The authors recognized that conventional input devices such as keyboards and mice limit intuitive interaction and proposed a contactless alternative using hand gestures captured via standard RGB cameras. Their work primarily addresses the non-linear challenges associated with gesture variability, including differences in hand orientation, finger articulation, background complexity, and lighting conditions.

Patel et al.[15](2024) proposed a real-time hand gesture recognition system integrated with a web-based application framework to enable intuitive human–computer interaction. The authors developed their system using Python for backend processing and JavaScript-based technologies for the frontend interface, ensuring seamless communication between gesture detection and user interaction modules. The core of their approach relies on MediaPipe, an open-source machine learning framework, which performs accurate hand tracking by identifying key landmark points and analyzing finger curvature and motion patterns. By leveraging webcam input, the system captures live hand gestures and classifies them using custom machine learning models trained on predefined gesture datasets. One of the primary strengths of the proposed framework is its real-time processing capability, which enables smooth and responsive gesture-based control without requiring specialized hardware. Additionally, the integration of a web interface makes the system platform-independent and accessible through standard browsers. However, the model's performance may be influenced by lighting conditions, background noise, and hand occlusion, which can impact landmark detection accuracy.

III. Proposed System

3.1 Overview

The suggested framework uses a web-based application to recognize movements in Indian Sign Language (ISL) and translate them into written and visual content. The system architecture consists of a browser-based avatar visualization module for animated sign representation, a Node.js-based backend for processing and storage, and a deep learning model for gesture detection. It prioritizes scalability, modularity, and device friendliness by offering an end-to-end solution framework.

3.2 System Modes

- **The Signer Mode:** It allows for fluent gesture-to-speech conversion and is targeted at sign language users. The system records hand motion with a camera feed and interprets it using a deep learning model to output text or voice in real time. Continuous signature is realized, granting fluent dialogue.
- **Mode of Listening:** This mode is designed for non-signers-or generic users-who want to understand the movements in sign language. It allows a hearing person to interact naturally with a signer by generating immediate voice or text output from the detected signals. For added clarity, the interface also displays an optional avatar or text visualization.
- **Practice/Learning Mode:** Through this guided mode, beginners can learn the motions of signs. These tasks include practice tasks, accuracy assessment of gestures, correctness feedback, and demonstrations through a visual avatar. This mode helps teachers and students become more proficient in ISL.

3.3 System Architecture

Figure 1: illustrates the system architecture for the real- time sign language to speech conversion framework. The architecture is composed of the following major components:

1) **Input Image:** The system, using the camera interface, captures a picture or a video frame of the user's hand gesture.
2) **CNN Model:** The frame captured acts as an input to a Convolutional Neural Network model, which then extracts features and recognizes the gesture.
3) **Classification Module:** The CNN output is fed into a classification layer that maps the gesture to the correct class of sign.
4) **Text Output Generation:** Once the gesture is classified, it is transformed into its textual meaning. This text serves as an intermediary representation for speech production and avatar rendering.
5) **User Display:** The front-end interface lets the user view textual output in real time for readability and verification purposes.
6) **Avatar Renderer:** An avatar-rendering module, receiving the identified text from the system in simultaneously, animates the matching sign providing visual feedback.

The combined workflow enables real-time recognition, translation, and visualization of sign language gestures, ensuring an intuitive and interactive user experience.

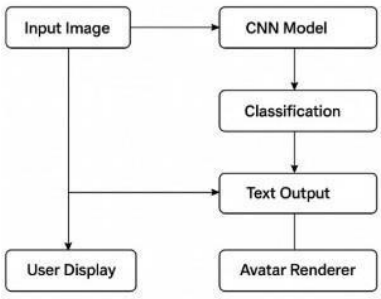

Fig. 1. System Architecture

The diagram below illustrates the design of the Real-Time Sign Language to Speech Conversion system. First, an Input Image is captured from a user's webcam. This feeds a CNN-based gesture recognition model that sends spatial information it has extracted from the image to a classification module for determining the relevant sign. The detected sign

is then converted to Text Output, which serves as the primary answer for further processing. The text is passed on to two parallel modules: one is the Avatar Renderer, which creates an animated visual representation of the detected sign, and the other is the User Display, showing the recognized motion in readable textual form. Real-time processing, modularity, and seamless integration among the deep learning model, rendering components, and user interface enable translation between visual motions and text and animated output.

3.4 Total System Latency Calculation

The system should process gesture inputs as fast as possible to allow real-time interaction. The total end-to-end latency L_{total} represents the cumulative delay caused by the CNN model, the classification module, the text-generation block, and the avatar-rendering pipeline.

The computation of the total delay is given by:

$$L_{total} = \max(L_{cnn}, L_{class}) + L_{text} + L_{render}$$

Where:

- L_{class}: classification and label mapping latency, 5–10 ms
- L_{cnn}: latency of CNN-based extraction for gesture features, 15–40 ms • L_{text} = interface update latency and text conversion (5–12 ms) • L_{render} = avatar animation and rendering time (10–20 ms)

Calculation example:

Ltotal = max(28ms, 7ms) + 9ms + 15ms = 52ms

This minimal latency ensures rapid and fluent sign-to-speech communication.

This method ensures the accuracy of the prediction by calculating a confidence score for each detected sign using a weighted mix of CNN output probability and gesture temporal stability.

$\text{GSI} = \alpha C_{cnn} + (1 - \alpha) C_{stability}$

Where:

- C_{cnn} = CNN softmax confidence probability
- $C_{stability}$ = frame-to-frame consistency score
- α = weighting factor in range $0 \leq \alpha \leq 1$ (default: 0.7)

A gesture is accepted only if

$\text{GSI} \geq \tau$

where is the adaptive confidence threshold of the system, which is 0.65 by default.

Fusion Layer combines the results of the CNN classifier and the confidence score model to generate more exact gesture predictions. It manages the following:

• Temporal smoothing of gesture sequences • Elimination of unclear or incomplete gestures • Stabilized label output for animation • Text results mapped to templates for avatar animation

The adaption rules prevent flickering caused by rapid frame-level changes and ensure smooth transitions within the animated avatar.

3.7 Output Generation Module

The final output of the system consists of two parallel streams:

1) **Text Output:** The identified sign will be reflected in the user interface after translation into the English language.
2) **Avatar Animation:** The text label allows the proper 3D avatar motion to be activated so that the sign may be visually represented.

For the purpose of facilitating system performance assessment and future improvements of continuous gesture sequence handling, each output is time-stamped and logged.

IV. Methodology

A. Summary

The proposed **Real-Time Sign Language to Speech Conversion** system is designed to recognize static and dynamic Indian Sign Language (ISL) gestures and convert them into speech in real-time. The framework integrates Convolutional Neural Networks (CNN) for gesture recognition, sequence modeling, and text-to-speech synthesis for spoken outputs. Additionally, a 3D avatar visualization module provides an interactive representation of recognized gestures. The system emphasizes low-latency processing, high accuracy, and usability across devices.

B. Data Collection and Preparation

The dataset comprises images and video frames of 26 ISL alphabets and common words, collected from multiple sources. Each sample was normalized to 64×64 pixels and augmented through rotation, flipping, and brightness variation to increase diversity and reduce overfitting. The preprocessed data is divided into 80% training, 10% validation, and 10% testing sets.

C. Gesture Recognition and Feature Extraction

The gesture recognition module uses a Convolutional Neural Network (CNN) with the following architecture:

- **Input Layer:** Accepts preprocessed gesture images.
- **Convolutional Layers:** Extract spatial features like edges and contours.
- **Pooling Layers:** Downsample feature maps to reduce computation.
- **Fully Connected Layers:** Classify gestures into 26 ISL classes.
- **Softmax Layer:** Provides probability distribution across all classes.

The CNN was trained using categorical cross-entropy loss and Adam optimizer, achieving high accuracy with minimal overfitting. Accuracy is calculated as:

$$\text{Accuracy} = \frac{N_{\text{correct}}}{N_{\text{total}}} \times 100\% \quad (1)$$

where N_{correct} is the number of correctly recognized gestures, and N_{total} is the total number of gestures.

Fig. 2. Performance Evaluation

Metric	Training Phase	Testing Phase
Accuracy	98.4%	96.2%
Precision	97.9%	95.4%
Recall	97.6%	95.1%
F1-Score	97.8%	95.2%

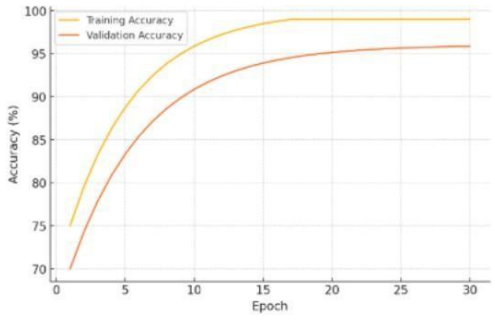

Fig. 3. Training and Validation Accuracy of CNN Model

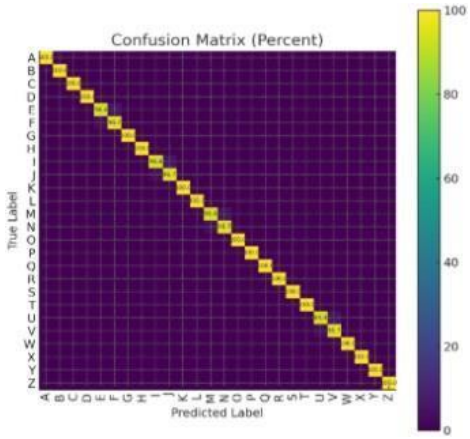

Fig. 4. Confusion matrix

D. Avatar Visualization

Recognized gestures are mapped into real-time visual feedback using a 3D avatar. The avatar uses a pure JavaScript animation library to show motions. This improves user engagement by adding visualization.

V. Results and Discussion

The experimental results prove that the proposed ISL-to-speech system works effectively and reliably for real-time gesture interpretation. While achieving accuracy, the CNN model attained a good generalization and stability with relatively slight misclassifications across visually identical motions. Because of the web-based deployment, the system can be deployed to run on a variety of devices without any extra software installation, while it also enhances accessibility and scalability. Finally, because of the inclusion of a 3D avatar with valuable visual feedback, the system is useful in both ISL learning and real-time communication.

Nevertheless, a number of limitations were identified: Dynamic gesture recognition, which requires motion and sequence analysis, is a key area for development since the existing system primarily deals with static indicators. Further additions of diverse backgrounds, skin types, and light variations in the dataset would enhance robustness by reducing bias. Despite these challenges, the system holds immense promise for a scalable and user-friendly ISL interpreter, while allow- ing for further development into more sophisticated gesture modeling.

VI. Conclusion

The proposed Avatar-Based ISL Toolkit provides a unified, easy-to-use approach toward real-time interpretation of Indian Sign Language. It bridges the gaps in communication between the hearing-impaired community and non-signers successfully by presenting an integrated framework comprising CNN-based gesture detection, scalable Node.js-MongoDB backend, and 3D avatar rendering. The toolkit is suitable for educational, assistive, and communication-oriented applications since great reliability can be observed in this model for static gestures recognized at an accuracy of 96.2

Future models, such as RNNs or 3D CNNs, can expand this system to identify dynamic gestures that include tem- poral movement patterns. This robustness will be increased and will generalize better by having more diverse datasets representing a range of skin tones, backdrops, and lighting situations. Further development, such as translation in both directions, speech/text to sign animations, will open further possibilities of practical usefulness; similarly, NLP for context- aware interpretation will add to it, while transfer learning optimization for mobile or edge devices will also help achieve that. This would also allow for gesture sequencing and facial expression detection to further enhance the avatar's expres- siveness, making organic and engaging ISL communication easier.

References

[1] M. Prabhakar et al., "Sign language conversion to text and speech," JETIR, vol. 9, no. 7, 2022.
[2] Y. Grover et al., "Sign language translation systems for hearing/speech impaired people: a review," ICIPTM, pp. 10–14, 2021.
[3] M. A. Alzubaidi, M. Otoom, and A. M. Abu Rwaq, "A novel assistive glove to convert Arabic sign language into speech," ACM TALLIP, vol. 22, no. 2, 2023.
[4] H. Monga et al., "Speech to Indian sign language translator," Recent Trends in Intensive Computing, pp. 9–15, 2021.
[5] P. Sharma et al., "Translating speech to Indian sign language using NLP," Future Internet, vol. 14, no. 9, 2022.
[6] J. Peguda et al., "Speech to sign language translation for Indian languages," ICACCS, pp. 1131–1135, 2022.
[7] Yin, K., Moryossef, A., Hochgesang, J., Goldberg, Y. and Alikhani, M., 2021, August. Including signed languages in natural language processing. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers) (pp. 7347-7360).

[8] N. Kahlon and W. Singh, “Machine translation from text to sign language: a systematic review,” UAIS, vol. 22, 2023.
[9] S. Ajay et al., “Indian sign language recognition using random forest classifier,” CONECCT, 2021.
[10] Alvin, A., Shabrina, N.H., Ryo, A. and Christian, E., 2021. Hand gesture detection for sign language using neural network with mediapipe. Ultima Computing: Jurnal Sistem Komputer, 13(2), pp.57-62..
[11] Ibarguren, A., Maurtua, I. and Sierra, B., 2010. Layered architecture for real time sign recognition: Hand gesture and movement. Engineering Applications of Artificial Intelligence, 23(7), pp.1216-1228.
[12] Oudah, M., Al-Naji, A. and Chahl, J., 2020. Hand gesture recognition based on computer vision: a review of techniques. journal of Imaging, 6(8), p.73.
[13] S. Stoll et al., “Text2Sign: Text-to-sign translation using NMT and 3D avatars,” CVPR Workshops, 2020.
[14] Hrishikesh, P., Akshay, V., Anugraha, K., TR, H.S. and Jyothisha, J.N., 2024. Vision based gesture recognition. Procedia Computer Science, 235, pp.303-315.
[15] Patel, M., Rao, S., Chauhan, S. and Kumar, B., 2024, December. Real-time Hand Gesture Recognition Using Python and Web Application. In 2024 1st International Conference on Advances in Computing, Communication and Networking (ICAC2N) (pp. 564-570). IEEE.