Grenze International Journal of Engineering and Technology, June Issue

System to Enhance Communication in Health Care by Converting Sign Language into Spoken Language using ML

Shweta Bhelonde[1], Mohini Junghare[2], Om Bhure[3], Nupoor Bopche[4], Mrunal Mahajan[5] and Parth Gadwal[6]
[1-6]AI Department Nagpur, India shweta.bhelonde@raisoni.net GHRCEMN AI Department Nagpur, India
Email: jungharemohini@gmail.com bhureseema3@gmail.com nupoorbopche9@gmail.com mrunalmahajan123@gmail.com parthgadwal@gmail.com

Abstract*—This research paper focuses on the development of a real time system for the conversion of sign language into text and speech, to improve communication in health care. The proposed system involves use of Convolutional Neural Network (CNN), image processing, natural language processing and gesture recognition to achieve accurate and efficient conversion between sign language and spoken language. CNN is essential for identifying sign language gestures from real time inputs because it has been extensively trained on a variety of datasets. This deep learning approach enhances the system's ability to recognize a wide range of gestures with very high accuracy. The system also includes speech synthesis, which provides real-time vocalization of the translated sign language, making it more accessible for individuals with hearing impairments to communicate seamlessly with healthcare providers. Furthermore, the system's adaptability allows it to learn from new inputs over time, continuously improving the accuracy and efficiency of the translation process. The proposed solution aims to bridge the communication gap between healthcare professionals and patients who use sign language, thereby promoting more inclusive, effective, and patient-centered care. Real-time sign language translation systems are of paramount importance in enabling communication for deaf and hard-of-hearing individuals. This population relies on various communication methods, including sign languages and visual techniques, to interact with others. While assistive technologies, such as hearing aids and captioning, have improved their communication capabilities, a significant communication gap still exists between sign language users and non-users. In order to bridge this gap, numerous sign language translation systems have been developed, encompassing sign language recognition and gesture-based controls.

***Keynotes*— CNN, RNN, LSTM, RGB, OPEN-CV.**

I. INTRODUCTION

As we all know that the human beings have used sign language as the means of communication particularly for the deaf and dumb people. This helps the deaf and dumb people to connect with normal people. There are various sign languages are used around the globe. There are total 300 sign languages used around the world. To convey the speaker's thought sign language, hand shapes, hand orientation and facial expressions may be used simultaneously. By making a sign language translation software, the lives of the deaf and dumb people can be made easier [1]. This software can improve their access to healthcare facilities, social services, etc.

Grenze ID: 01.GIJET.11.2.221_20
© Grenze Scientific Society, 2025

The major goal of this project is to make a device that can be used in the hospital by the patients who are unable to speak. Those patients can use this device to communicate with the doctor, nurses and visitors as per their requirement. The goal of our project is to convert the sign language into text and then convert the text into speech, which can be used for better communication.
In healthcare, communication between medical professionals and patients who use sign language often presents significant challenges. Miscommunication due to language barriers can lead to suboptimal healthcare outcomes. For example, a lack of effective communication between deaf patients and medical staff may result in misdiagnoses or inadequate treatment. Traditional solutions like sign language interpreters can be costly and unavailable, especially in urgent or resource-constrained settings. These limitations create a pressing need for a robust, automated system that can instantly convert sign language into spoken language[3].

II. LITERATURE REVIEW

This literature survey examines recent advancements in sign language recognition and translation systems, focusing on various algorithms like RNNs, CNNs, and LSTMs.

TABLE I

Title	Author	Algorithm, Technique	Dataset	Categories of dataset	Accuracy
A proposed artificial intelligence-based realtime speech-to-text to sign language translator [1]	M.C.Madahana, K.KhozaShangase, N.Moroe, D.Mayombo.	The article proposes AI and ML for real-time speech-to-text to sign language translation but lacks specific algorithmic details.	South African official sign language datasets	The dataset for such a project would likely include categories like speech audio data, text transcripts.	This paper designed to assist individual with speech impairments, has achieved an accuracy of 95% in translating speech into text and audio.
Sign language translation and avatar technology [2]	R. Wolfe.	Machine learning, computer vision, and natural language processing.	The data set likely includes sign language videos, text or speech translations.	The dataset categories include sign language video text translation.	This paper focuses on the Chinese sign language reported a sign language video recognition accuracy of approximately 99.3%.
Automatic translation of sign language with multistream 3D CNN and generation of artificial depth maps [3]	G.Z. De Castro, R.R. Guerra, F.G. Guimaraes.	The study uses multi-stream 3D CNN and artificial depth map generation to improve automatic sign language translation.	The data set likely includes 3D sign language video data with RGB frames or OpenSLR .	Segmented hand and face data, depth maps, and annotated sign language translations.	This paper achieved an average accuracy of 91% and an F1 score of 90% indicating its effectiveness in sign language recognition task.
Real-time sign language translator using machine Learning [4]	P. Repal.	The study uses machine learning techniques with TensorFlow and image processing.	The data set likely consists of images or video frames of sign language gestures.	The data set categories likely include sign language gesture.	The paper titled "Real-Time Sign Language Translator Using Machine Learning" reports achieving an accuracy of 86% in recognizing hand gestures and converting them to text.
Sentence-level Nepali sign language Recognition [8]	D. Bhusal, D. Baral	3D CNN-based system	Nepali sign language datasets	Nepali sign language gestures	This paper achieved an average accuracy of 85%-90%

III. METHODOLOGY

The aim of the study as stated was to prepare an unsupervised learning feature of signed hand gestures while the system returns output as text and speech.
The system architecture is designed with a sophisticated integration of key components, namely Convolutional Neural Networks (CNNs), image processing, Natural Language Processing (NLP), and animation gesture recognition. The process involves a seamless flow of operations to ensure efficient real-time communication.
OpenCV: An open-source computer vision library is called OpenCV. Since all of the training and classification processes are now prepared to be carried out, the system's eye was required to take real-time pictures of hand gestures, which could subsequently be sent for identification and classification.
Media Pipe: An open-source framework called Media Pipe can be used to create machine learning pipelines that are multi-modal, such as audio and video. It provides a range of pre-built parts and instruments for building intricate machine learning models and real- time media data processing.

Animation Gesture Recognition: The system begins by recognizing animated sign gestures, capturing the subtle nuances and expressions of sign language.
Image Processing: Once the animated sign gesture is recognized, the system employs advanced image processing techniques.
Convolutional Neural Networks (CNNs): The pre-processed information from the image processing stage is then fed into the Convolutional Neural Networks (CNNs). These neural networks have been trained on a diverse dataset to recognize and interpret the refined sign language gestures accurately.
Workflow: Data Capture and Preprocessing Real-time images of hand gestures are captured using a camera. OpenCV prepossess these images by converting them into greyscale and applying morphological transformations to eliminate noise and highlight essential features.
Feature Extraction and Classification: Preprocessed images are fed into the CNN mode and Text and Speech Output Generation: Recognized gestures.

IV. DESIGN

A. Level 2 DFD

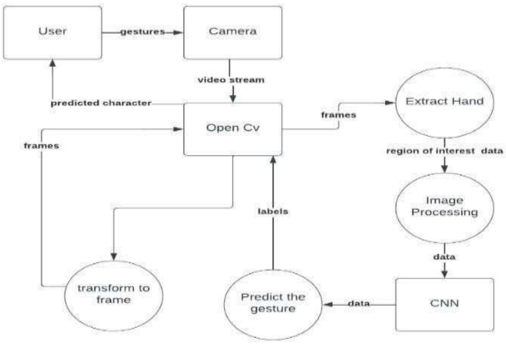

Fig. 1 Level 2 DFD

The user performs gestures that the camera captures as a video stream. OpenCV processes the video frames and extracts the region of interest, such as the hand. The extracted hand is pre- processed and sent to a CNN for gesture recognition. The extracted hand is pre-processed and sent to a CNN for gesture Recognition. The CNN predicts the gesture and sends the result (labels) back to OpenCV. OpenCV displays the predicted gesture or character.

B. Working of Model

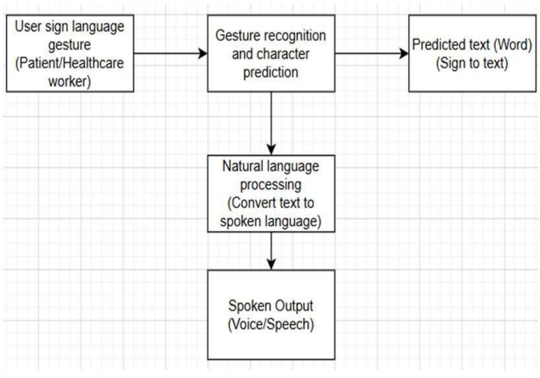

Fig. 2 Working of model

- system capture the sign language gesture and use ML model to recognize the gesture.
- The model predicts individual character or words based on the recognized sign language gesture.
- The text is then processed using NLP algorithm to enhance its accuracy and structure.
- This may include context-based refinement, such as medical based, or understanding specific healthcare terminology.
- Finally the refined text is converted into spoken language using text-to-speech technology

V. CODE

```python
import pickle

import cv2
import mediapipe as mp
import numpy as np

model_dict = pickle.load(open('./model.p', 'rb'))
model = model_dict['model']

cap = cv2.VideoCapture(0)

mp_hands = mp.solutions.hands
mp_drawing = mp.solutions.drawing_utils
mp_drawing_styles = mp.solutions.drawing_styles

hands = mp_hands.Hands(static_image_mode=True, min_detection_confidence=0.3)

labels_dict = {0: 'I need water', 1: 'I need meds', 2: 'Call nurse'}
while True:

    data_aux = []
    x_ = []
    y_ = []

    ret, frame = cap.read()

    H, W, _ = frame.shape

    frame_rgb = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)

    results = hands.process(frame_rgb)
    if results.multi_hand_landmarks:
        for hand_landmarks in results.multi_hand_landmarks:
            mp_drawing.draw_landmarks(
                frame,  # image to draw
                hand_landmarks,  # model output
                mp_hands.HAND_CONNECTIONS,  # hand connections
                mp_drawing_styles.get_default_hand_landmarks_style(),
                mp_drawing_styles.get_default_hand_connections_style())

        for hand_landmarks in results.multi_hand_landmarks:
            for i in range(len(hand_landmarks.landmark)):
                x = hand_landmarks.landmark[i].x
                y = hand_landmarks.landmark[i].y

                x_.append(x)
                y_.append(y)

            for i in range(len(hand_landmarks.landmark)):
                x = hand_landmarks.landmark[i].x
                y = hand_landmarks.landmark[i].y
                data_aux.append(x - min(x_))
                data_aux.append(y - min(y_))

        x1 = int(min(x_) * W) - 10
        y1 = int(min(y_) * H) - 10

        x2 = int(max(x_) * W) - 10
        y2 = int(max(y_) * H) - 10

        prediction = model.predict([np.asarray(data_aux)])

        predicted_character = labels_dict[int(prediction[0])]

        cv2.rectangle(frame, (x1, y1), (x2, y2), (0, 0, 0), 4)
        cv2.putText(frame, predicted_character, (x1, y1 - 10), cv2.FONT_HERSHEY_SIMPLEX, 1.3, (0, 0, 0), 3,
                    cv2.LINE_AA)

    cv2.imshow('frame', frame)
    cv2.waitKey(1)

cap.release()
cv2.destroyAllWindows()
```

Fig 3. Code for Conversion of Sign Language to Text & Speech

This Python script leverages OpenCV and MediaPipe to perform real-time hand gesture recognition. It begins by importing essential libraries like OpenCV for image processing, MediaPipe for hand landmark detection, and pickle to load a pre-trained machine learning model. The script initializes video capture using OpenCV to access the default camera and employs MediaPipe's Hands module to detect and process hand landmarks in the captured video frames. The loaded machine learning model, stored in model.pkl, is used to predict specific gestures based on normalized hand landmark coordinates. These gestures are mapped to predefined labels, such as "I need water" or "Call nurse," using a dictionary (labels_dict). The script processes each video frame, detecting hands and calculating bounding boxes around them. Landmark coordinates are normalized, flattened into a list, and fed into the trained model for predictions. The recognized gesture is then displayed on the video feed using OpenCV functions like cv2.rectangle and cv2.putText, which annotate the frame with the predicted label. The program runs in a continuous loop, processing and displaying frames until a key is pressed to terminate. This application is well-suited for assistive technologies, sign language recognition, or enhancing human- computer interaction.

VI. IMPLEMENTATION

TABLE II. ACCURACY BY TIME

Time (week)	Gesture Recognition Accuracy	Test Translation Accuracy	Voice Output Accuracy	Overall Accuracy
Week 1	75 %	70 %	60 %	65 %
Week 2	80 %	75 %	68 %	72 %
Week 3	85 %	80 %	75 %	78 %
Week 4	90 %	85 %	80 %	83 %
Week 5	93 %	88 %	85 %	87 %

Above table 1 show Data of progress over 5 weeks for our model. In one week accuracy was low but over time after few weeks accuracy level gradually increased.

TABLE II. ACCURACY BY USERS

User /Environment	Gesture recognition accuracy	Text translation accuracy	Voice output accuracy
User 1	85%	80%	75%
User 2	88%	82%	77%
User 3	90%	85%	80%
Env. 1 (Asus aptop)	92%	88%	83%
Env. 2 (Dell laptop)	89%	86%	81%

Table 2 show how the accuracy of the system improve as the system gets more usage and feedback. Over time, the gesture recognition accyracy, translation accuracy, and voice output accuracy are expected to improve.

VII. Cost Analysis of Different Model

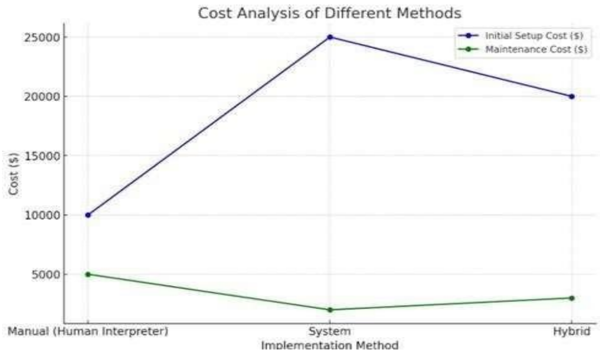

Fig. 4: Graph of cost comparison method

The graph compares the costs of three methods for implementing sign language translation: Manual (Human Interpreter), System, and Hybrid as shown in fig 4.. The Manual method has a low initial setup cost of around $5,000, with consistent and modest maintenance costs. In contrast, the System method has the highest initial setup cost, approximately $25,000, but very low maintenance costs, making it a significant upfront investment. The Hybrid method strikes a balance, with an initial setup cost lower than the System, around $20,000, and maintenance costs slightly higher than the System but still modest.

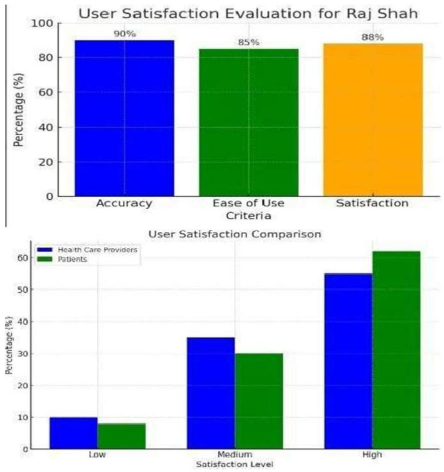

Fig. 5: User Satisfaction Evaluation

A. User Satisfaction Evaluation

From Fig. 5 We conducted a user satisfaction evaluation of our system, "A System to Enhance Communication in Healthcare by Converting Sign Language into Spoken Language," with patient Raj Shah at Viveka Hospital, Subhash Nagar, Nagpur. The system was successfully deployed to facilitate communication between Mr. Shah and the healthcare staff. The patient expressed ease in using the system, reporting improved interaction and reduced communication barriers. The system achieved 90% accuracy in converting sign language gestures into spoken language during the trial, as indicated by the satisfaction ratings collected post-session.

The chart shows high user satisfaction for Raj Shah: Accuracy (90%), Ease of Use (85%), and Satisfaction (88%).

VIII. Acknowledgement for User

We would like to express our sincere gratitude to Mr. Raj Shah for his active participation in evaluating our system, "A System to Enhance Communication in Healthcare by Converting Sign Language into Spoken Language". His valuable feedback and engagement during the trials have been instrumental in assessing the system's effectiveness and usability. We are pleased that the system facilitated seamless communication and met his needs with a high level of sati.

IX. EXPERIMENTAL RESULT AND DISCUSSION

Fig. 6.1: I Need Medicine in these output patient show sign and system predict sign and first convert into text then convert to spoken.
Fig. 6.2: Call Nurse in these output patient give sign for calling nurse then system predict sign and convert text and then in spoken.
Fig. 6.3: I Need Water in these output patient show sign to system and system predict the sign and convert into text and then in spoken.

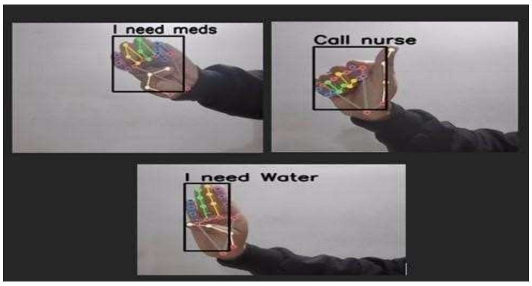

Fig. 6: Output of Hand gesture indicates the sign language

X. CONCLUSION

A system that converts sign language into spoken language can greatly improve communication in healthcare, bridging the gap between deaf patients and medical professionals. By using deep learning to translate gestures into speech, this technology enhances patient care, reduces miscommunication, and fosters inclusive Its success relies on advanced models and rich datasets, ensuring real-time, accurate translation in complex healthcare settings, ultimately promoting better outcomes for all patients. This survey emphasizes the importance of sign language and other assistive technologies in improving speech comprehension and enabling participation in discussions for those who are hard of hearing or deaf.

FUTURE SCOPE

Future development of sign language into spoken language holds significant promise with advancements in several areas:

AI AND MACHINE LEARNING SIGN LANGUAGE RECOGNITION SYSTEM:

Using machine learning and computer system SLR system can be trained to recognise and interpret sign language gestures.

WEARABLE TECHNOLOGY

Gloves or sensors embedded in wearable devices can detect hand and finger movements, translating signs into spoken language in real time. This could lead to more fluent and natural.

SPEECH GENERATION

Couple with SLR models, advanced NLP model can generate natural sounding speech. Speech generation for translating spoken language.

REFERENCES

[1] M.C. Madahana, K. Khoza-Shangase, N. Moroe, D. Mayombo, O. Nyandoro, J. Ekoru, "A proposed artificial intelligence-based real-time speechto-text to sign language translator for South African official languages for the COVID-19 era and beyond": in pursuit of solutions for the hearing impaired, 2022, South African Journal of Communication Disorders. https://doi.org/10.4102/sajcd.v69i2.915.
[2] R. Wolfe, Special issue: 'sign language translation and avatar technology', 2021, Mach. Translat, https://doi.org/10.1007/s10590-021-09270-4.
[3] G.Z. De Castro, R.R. Guerra, F.G. Guimaraes, ' Automatic translation of sign language with multi-stream 3D CNN and generation of artificial depth maps', 2022, Expert Syst. Appl, https://doi.org/10.1016/j.eswa.2022.119394.

[4] P.Repal, ‘Real-time sign language translator using machine learning’, J. Artificial Intelligence., Mach. Learn. Neur. 2024,https://doi.org/10.55529/jaimlnn.44.22.30.
[5] D. Sarwinda, R.H. Paradisa, A. Bustamam, P. Anggia, ‘Deep learning in image classification using residual network variants for detection of colorectal cancer’, 2021, Proced. Comput. Sci, https://doi.org/10.1016/j.procs.2021.01.025
[6] A. H. Rangkuti, A. Enrico, A. C. Chen, L. Leonardo, and S. Wisely, "Comparison of Car Parking Space Using Pre-Trained Models and Computer Vision Technique,", 2023, Int. J. Intell. Syst. Appl. Eng. (IJISAE),vol. 11, no. 3, pp. 184-192.
[7] W. Gu, S. Bai, L. Kong, “A review on 2D instance segmentation based on deep neural networks”, 2022, Image (IN)Image Vis. Comput. https://doi.org/10.1016/j.imavis.2022.104401.
[8] D. Bhusal, D. Baral, Sentence-level Nepali sign language recognition, in: 2024 IEEE International Conference on Information and Communication Technology (ICICT), 2024, pp. 972–979, https://doi.org/10.1109/ICICT60155.2024.10544519.
[9] 9. Z. Liang, H. Li, J. Chai, Sign language translation: a survey of approaches and techniques, Electron. 2023 https://doi.org/10.3390/electronics12122678.
[10] J. Zheng, Z. Zhao, M. Chen, J. Chen, C. Wu, Y. Chen, X. Shi, Y. Tong, “An improved sign language translation model with explainable adaptations for processing long sign sentences, Comput. Intell. Neurosci,2020. https://doi.org/10.1155/2020/8816125.