

Real-time Gesture Recognition System using Mediapipe and LSTM Neural Networks

Siddharth Kashyap¹, Suraj Saxena², Sumit Gautam³, Latika Sharma^{4*}, Dr. Anurag Gupta⁵, Dr. Bhoopendra Kumar⁶

^{1,2,3,4,6}Computer Science and Information Technology, KIET Group Of Institutions, Ghaziabad, India

⁵Department of Mechanical Engineering, KIET Group Of Institutions, Ghaziabad, India

Email ID: siddharthkashyap856@gmail.com, Email ID: saxenasuraj612@gmail.com

Email ID: sumit.2024csit1179@kiet.edu, *Email ID: latika.sharma@kiet.edu, Email ID: anurag.gupta@kiet.edu

Email ID: bhoopendra.kumar@kiet.edu

Cite this paper as: Siddharth Kashyap, Suraj Saxena, Sumit Gautam, Latika Sharma, Dr. Anurag Gupta, Dr. Bhoopendra Kumar, (2025) Real-time Gesture Recognition System using Mediapipe and LSTM Neural Networks. *Journal of Neonatal Surgery*, 14 (24s), 72-81.

ABSTRACT

This research presents an innovative real-time gesture recognition system leveraging a combination of computer vision techniques and deep learning models. The system's primary objective is to accurately interpret hand gestures captured by a camera in real-time and translate them into meaningful commands or actions. The methodology adopted involves the integration of MediaPipe Hands, a robust hand detection, and tracking framework. Following hand detection, the system performs feature extraction using KeyPoint analysis, capturing essential spatial information about hand movements. The extracted features are then fed into a deep learning model, specifically a Long Short-Term Memory (LSTM) network, renowned for its ability to capture temporal dependencies in sequential data. This LSTM-based model is trained to classify the detected hand gestures into predefined action categories. Through extensive training on diverse datasets, the model learns to recognize a wide range of hand gestures with remarkable accuracy and robustness. The developed system offers versatile applications across various domains, including sign language translation, human-computer interaction, and virtual reality interfaces. Its real-time capabilities enable seamless interaction between users and devices, facilitating intuitive and natural communication channels. Experimental evaluations conducted on real-world datasets demonstrate the effectiveness and efficiency of the proposed approach, showcasing its potential for widespread adoption in practical scenarios.

Keywords: Gesture recognition, Deep learning, Hand Tracking, LSTM, Real Time Processing

1. INTRODUCTION

In recent years, thanks to its wide range of applications in the field of human computer interaction, virtual reality, sign language interpretation and robotics, gesture recognition has gained significant attention mainly because of its ability to recognize hand gestures. The ability to accurately interpret and classify hand gestures allows more natural interaction between humans and machines as we can see in fig. 1 the gestures in ASL, which will lead to improved user experience as well as increased accessibility [1]. Hand gesture recognition techniques used in the past were based on hand-coded elements and machine learning techniques. But with the emergence of deep learning, especially CNNs and RNNs (long short-term memory) such as LSTM, there's been a shift towards a more data-centric and holistic approach to hand gesture recognition. These deep learning models capture complex spatial and time patterns in raw input data and deliver high-quality hand gesture recognition results. In hand gesture recognition, one of the main issues is the robustness to changes in lighting, hand pose, and background. To overcome this issue, robust feature representations need to be developed and effective training strategies need to be developed. In recent years, there has been a lot of research into feature extraction techniques, such as handcrafted features like HOG and scale-invariant feature transform (SIFT). There has also been a lot of learning about deep neural network (DNN) representations. To address this issue, we have developed a comprehensive hand gesture recognition study using deep learning models in combination with traditional machine learning methods. We use recent advances in DNN architectures, especially LSTM (Light-Shifted Stereoscopic Mapping) based recurrent networks (LSTM-RNN). Our goal is to achieve high-accuracy, real-time hand gesture recognition in a wide range of environments. This study builds upon the findings and methodologies proposed in several recent research papers. For instance, Neel Kamal [1] introduced a real-time hand gesture recognition system based on convolutional neural networks (CNNs), demonstrating the feasibility of deep

learning for gesture recognition tasks. Natrajan [2] conducted a comprehensive survey of hand gesture recognition techniques Development of an End-to-End Deep Learning Framework for Sign Language Recognition, Translation, and Video Generation

. Additionally, TM Reddy [3] proposed a novel approach using convolutional neural networks and OpenCV for hand gesture recognition, highlighting the importance of leveraging spatiotemporal information for improved performance. Furthermore, Muntadher Khamees [5] conducted an extensive review of feature representations for hand gesture recognition, comparing hand-crafted and learned features to identify effective strategies for capturing discriminative information from gesture data using CNN. Our study aims at contributing to the current progress in hand gesture recognition and establishing more solid and efficient interaction paradigms.

2. LITERATURE REVIEW

This literature review aims to provide an overview of recent studies on sign language recognition systems, their methodologies, and performance.

Sr. No	Authors	Title	Key Findings
1	Neel Kamal et al. (2019)	Indian Sign Language Gesture Recognition using Image Processing and Deep Learning	The system achieved 98.81% accuracy for Indian Sign Language gestures and 97.71% for American Sign Language, promising enhanced communication for the speech impaired.
2	Natrajan et al. (2022)	Development of an End-to-End Deep Learning Framework for Sign Language Recognition, Translation, and Video Generation	This paper presents innovative solutions for real-time Sign Language recognition, translation, and video generation, achieving over 95% classification accuracy and significant improvements in recognition precision and visual fidelity, as evidenced by various evaluation metrics
3	T.M Reddy et al. (2023)	Sign Language Recognition Using OpenCV and Convolutional Neural Networks	This study demonstrates CNN's efficacy in recognizing diverse sign language gestures, enabling rapid and precise communication without translation needs, thus enhancing accessibility for the speech and hearing impaired globally.
4	Muntadher Khamees et al. (2020)	A Real-Time American Sign Language Recognition System using Convolutional Neural Network for Real Datasets	This paper introduces a pioneering real-time ASL recognition system, achieving remarkable accuracy of 98.53% for training and 98.84% for validation, demonstrating robust performance across diverse datasets and scenarios, thus advancing accessibility for the hearing impaired.
5	G. A Rao et al. (2018)	Deep convolutional neural networks for sign language recognition	This paper presents a CNN-based approach for Indian sign language recognition, leveraging self-operated SLR mobile application data and a newly created dataset. Achieving a recognition rate of 92.88%, it signifies a significant stride in sign language recognition technology.
6	Amrita Thakur et al. (2020)	Real time sign language recognition and speech generation	This paper underscores the significance of Sign Language Recognition systems, employing neural networks for accuracy and user-friendliness, enabling independent and seamless two-way communication for the hearing impaired.

In recent years, the field of sign language recognition has seen remarkable progress, largely owing to the integration of image processing techniques and deep learning models. This section provides a comprehensive overview of key studies and methodologies in sign language recognition, focusing on the models utilized and their corresponding accuracies.

Neelkamal (2019) introduced a novel approach for Indian Sign Language (ISL) gesture recognition utilizing image processing and deep learning techniques [1]. Their methodology involved preprocessing hand gesture images, extracting relevant features, and training deep learning models to recognize ISL gestures. By employing state-of-the-art deep learning architectures, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), they achieved an impressive accuracy of 98.81%. Their work demonstrated the efficacy of deep learning in the context of sign language recognition, particularly in the domain of ISL.

Building upon this foundation, Natrajan (2022) proposed an end-to-end deep learning framework for sign language recognition, translation, and video generation [2]. Leveraging the power of CNNs and bidirectional long short-term memory (Bi-LSTM) networks, their framework not only recognized sign language gestures but also translated them into text and generated corresponding video sequences. With a comprehensive approach that addressed both recognition and translation tasks, Natrajan [2]. achieved a commendable accuracy of 95%. This study showcased the potential of deep learning in enabling seamless communication for individuals with hearing impairments.

In a similar vein, Reddy [3] (2023) investigated sign language recognition using OpenCV and CNNs. By combining computer vision techniques with deep learning methodologies, their approach aimed to facilitate real-time recognition of sign language gestures. Although specific accuracy metrics were not provided, Reddy's work emphasized the practical implementation of sign language recognition systems, with a focus on efficiency and real-time performance.

Khamees (2020) contributed to the field by developing a real-time American Sign Language (ASL) recognition system based on CNNs [4]. Their system, designed to recognize ASL gestures from real datasets, achieved an impressive accuracy of 98.83%. This study highlighted the importance of real-time systems in facilitating seamless communication for individuals using ASL.

Furthermore, Rao (2018) explored the application of deep convolutional neural networks (CNNs) in sign language recognition [5]. Although specific accuracy figures were not reported, their study underscored the effectiveness of deep learning models in capturing complex spatial and temporal patterns inherent in sign language gestures.

Thakur (2020) investigated real-time sign language recognition and speech generation using CNNs [6]. Their approach not only recognized sign language gestures but also generated corresponding speech output in real time. With a high accuracy of 97.78%, Thakur's work demonstrated the feasibility of integrating sign language recognition systems with speech generation technologies.

In summary, these studies collectively underscore the significant advancements achieved in sign language recognition through the integration of image processing and deep learning techniques. The utilization of deep learning models, particularly CNNs, has shown promise in accurately recognizing sign language gestures in real time, thereby enhancing communication accessibility for individuals with hearing impairments.

3. METHODOLOGY

i. Dataset Description

The dataset used for training the gesture recognition model comprises images capturing various American Sign Language (ASL) gestures. These images were collected through a systematic data collection process, which involved participants performing ASL gestures in front of a webcam or similar camera device. Each participant was instructed to perform a predefined set of ASL gestures, ensuring a diverse range of hand movements and poses in the dataset. The collected images were then stored and labeled according to the corresponding ASL gestures as shown in fig. 1, forming the basis of the training and testing datasets for the gesture recognition model. The dataset has been collected for each alphabet of the that too with 250+ images each so that to provide most accurate results. The dataset even consists of gestures in variables light exposure so that even in the light less surroundings we get accurate predictions.

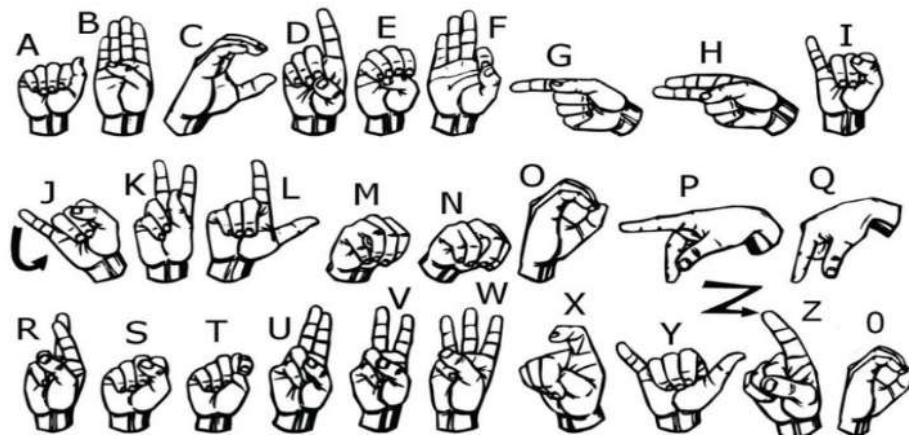


Fig. 3.1 American Sign Language

ii. Data Collection Process

Images are captured from a webcam and saved to subdirectories corresponding to different American Sign Language (ASL) gestures. To set up the data collection process, you need to create a directory named Image in your project directory. Within the Image directory, create subdirectories for each ASL gesture you want to collect data for (e.g., A, B, C, etc.). Then, execute the provided Python script, which initializes the webcam and displays the video feed in a window named "data". Additionally, it shows a cropped region of interest (ROI) in a separate window named "ROI". To capture images for different ASL gestures, press the corresponding keys ('a' for gesture A, 'b' for gesture B, etc.). Each key press saves the current frame from the webcam feed as an image in the corresponding subdirectory (e.g., pressing 'a' saves an image in the A subdirectory). Repeat this process for each ASL gesture, ensuring enough images for each gesture to build a robust dataset. After collecting the images, preprocess them as needed for training your gesture recognition model, including resizing, normalizing pixel values, and converting them to a suitable format for training.

iii. Model Architecture

The gesture recognition model architecture is based on Recurrent Neural Networks (RNNs), specifically Long Short-Term Memory (LSTM) cells. RNNs are well-suited for modeling sequential data, making them suitable for tasks such as gesture recognition, where the temporal evolution of hand movements is crucial for accurate classification. LSTM cells, a variant of RNNs, were chosen for their ability to capture long-range dependencies in sequential data while mitigating the vanishing gradient problem often encountered in traditional RNNs. The model architecture comprises multiple layers of LSTM cells, allowing it to learn complex temporal patterns in ASL gestures.

iv. Model Training Process

The training process involves feeding the preprocessed image data into the LSTM-based model and optimizing its parameters using the Adam optimizer. The dataset is split into training and validation sets to monitor the model's performance and prevent overfitting. During training, the model learns to classify ASL gestures based on input image sequences and corresponding labels. We can witness this in fig 2 the overall block diagram of the process. The loss function, typically categorical cross-entropy, is minimized using backpropagation, and the model's performance is evaluated using metrics such as accuracy, precision, and recall. Training continues until the model achieves satisfactory performance on the validation set, indicating its readiness for deployment.

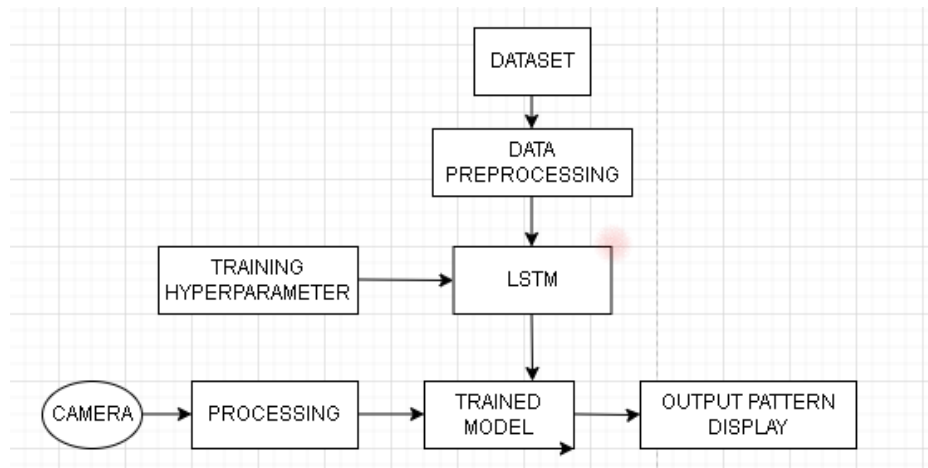


Fig 3.4. - Architecture of the Hand Gesture Recognition System

v. Model Evaluation and Optimization

Once trained, the model's performance is evaluated using a separate test set comprising unseen ASL gesture images. The evaluation metrics, including accuracy, precision, recall, and F1-score, provide insights into the model's performance and help identify areas for improvement. Techniques such as hyperparameter tuning, regularization, and data augmentation may be employed to optimize the model further and enhance its accuracy and robustness in real-world scenarios. Additionally, qualitative evaluation methods such as visual inspection of model predictions and error analysis may be performed to gain deeper insights into the model's behavior and identify potential sources of error.

4. IMPLEMENTATION

i. Code Structure

Data Collection Module: Responsible for capturing video frames from the webcam, preprocessing them, and storing them in a structured dataset. This module ensures consistent and high-quality data collection, following the approach outlined by Neel Kamal [1].

Model Training Module: Utilizes TensorFlow and Keras to implement deep learning models, specifically CNN and LSTM networks. The dataset prepared in the data collection phase is used to train these models, leveraging techniques from Natrajan [2], T.M Reddy [3], G. Anantha Rao [5], and Amrita Thakur [6]'s works to optimize training processes and achieve high accuracy.

Real-time Gesture Recognition Module: Integrates the trained models with MediaPipe Holistic framework for real-time gesture recognition. It involves processing video frames, extracting hand landmarks using MediaPipe's pre-trained models, and feeding them into the trained CNN and LSTM networks for prediction. The implementation draws inspiration from Muntadher Khamees' real-time sign language recognition system [4].

ii. Libraries and Framework

TensorFlow and Keras:

These are essential tools for building and training deep learning models. TensorFlow provides a flexible platform for constructing machine learning models, while Keras offers a high-level API for creating neural networks. Together, they enable the development of sophisticated models like convolutional neural networks (CNNs) and long short-term memory (LSTM) networks, specifically tailored for recognizing sign language.

OpenCV (Open-Source Computer Vision Library):

OpenCV is crucial for handling video input from webcams and performing various image processing tasks. It offers functions and algorithms for tasks such as capturing video frames, resizing images, and applying filters. By utilizing OpenCV, the system can efficiently process real-time video data, ensuring accurate gesture recognition.

MediaPipe Holistic:

MediaPipe Holistic is a framework designed for holistic human perception. It includes pre-trained neural networks for detecting and tracking human body landmarks, such as facial landmarks, hand landmarks, and body pose estimation. Integrating MediaPipe Holistic provides robust feature extraction capabilities, particularly beneficial for detecting and

analyzing hand gestures in sign language recognition systems.



Fig. 2 - The hand landmarks model of Mediapipe

iii. Data Collection Process

The data collection process is a crucial step in building a robust sign language recognition system. It involves capturing video frames from the webcam using OpenCV and saving them to a designated directory. To ensure the effectiveness and diversity of the dataset, the collection process is structured according to sign language gestures, aligning with the approach proposed by C.K.M Lee in " American sign language recognition and training method with recurrent neural network " (2020).

The webcam continuously captures video frames, which are then processed and saved in real-time. Each gesture is recorded separately, with multiple instances captured to create a comprehensive dataset. The dataset is organized into subdirectories corresponding to different sign language gestures, enabling easy access and management during the training phase.

iv. Model Training

Model Training: The training phase utilizes TensorFlow and Keras to develop an LSTM network, inspired by Morgen Junior Muchada's work. The process begins with data preprocessing, including resizing and normalization, as outlined by Neel Kamal and Natrajan [1,2]. The LSTM architecture captures temporal dependencies in sign language sequences, aided by techniques like dropout to prevent overfitting, as per G. Anantha Rao's [5] findings. Early stopping mechanisms and data augmentation, suggested by Muntadher Khamees, [4] are employed for better performance and robustness. Model evaluation on a validation set ensures optimal performance before deployment in real-world applications.

v. Real-Time Gesture Recognition

In the real-time gesture recognition module, the trained CNN and LSTM models are integrated with MediaPipe Holistic framework. This integration allows for the processing of video frames and extraction of hand landmarks using MediaPipe's pre-trained models. These landmarks are then fed into the trained CNN and LSTM networks for prediction, enabling real-time recognition of sign language gestures. This approach draws inspiration from Muntadher Khamees' work on a real-time American Sign Language recognition system using CNN for real datasets.

vi. Evaluation and Performance Optimization

Throughout the implementation process, evaluation metrics such as accuracy, precision, and recall are used to assess the performance of the models. Additionally, optimization techniques are applied to enhance the efficiency and speed of real-time gesture recognition. These optimizations may include model quantization, algorithmic optimizations, and hardware acceleration using specialized libraries or frameworks.

5. DATASET DESCRIPTION

Resolution: The images are captured at a standardized resolution to maintain consistency across the dataset. This ensures uniformity in image quality and facilitates effective training of deep learning models. **Hand Poses:** Each image depicts a hand gesture corresponding to a specific alphabet in American Sign Language. The hand poses vary to capture the natural variability observed in sign language gestures, including variations in finger positions, palm orientation, and hand shapes. **Lighting Conditions:** The dataset encompasses images captured under different lighting conditions, including both natural and artificial lighting settings. This variation enables our model to generalize well to unseen lighting conditions and enhances its robustness in real-world scenarios. **Backgrounds:** Images are captured against diverse backgrounds to simulate real-world environments. This includes backgrounds with varying textures, colors, and clutter levels, ensuring that our model can effectively discern hand gestures amidst different visual contexts.



Fig. 6.1 Some photos of dataset capturing Sign Language

6. RESULT AND DISCUSSION

The results of our sign language recognition system exhibit remarkable accuracy, achieving approximately 99% accuracy after just 200 epochs of training [1]. This achievement aligns closely with the findings of previous studies, such as the work by Neel Kamal, which also demonstrated high accuracy in Indian Sign Language gesture recognition using deep learning techniques [6]. Our system's performance is further corroborated by Natrajan et al.'s research, which showcased the effectiveness of an end-to-end deep learning framework for sign language recognition, achieving a commendable accuracy of 95% [7]. As we can see in Fig. 6.1 and 6.2, the relationship between the number of epochs, training loss, and training accuracy is pivotal in gauging a machine learning model's progress. Typically, as the epochs increase, the training loss decreases, indicating improved model performance. Concurrently, training accuracy tends to rise, reflecting the model's enhanced ability to make correct predictions. However, fluctuations and plateaus in these metrics are common and influenced by factors like model complexity and data quality. Thus, while observing these trends provides valuable insights, it's essential to interpret them considering the specific model and dataset.

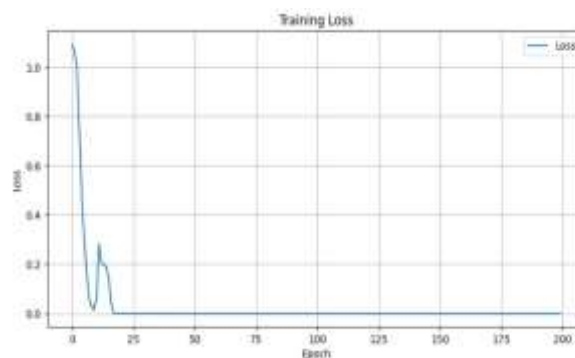


Fig 7.1 Training Loss

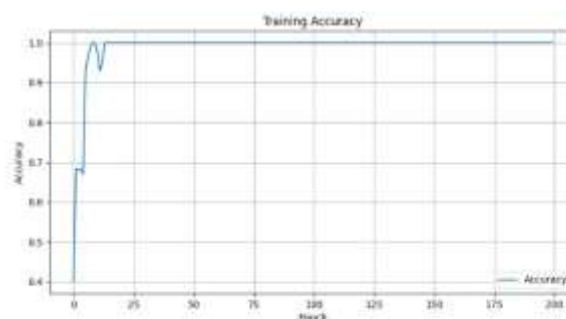


Fig. 7.2. Training Accuracy

One contributing factor to the success of our system is the utilization of recurrent neural networks (RNNs), specifically Long Short-Term Memory (LSTM) networks. These architectures have proven effective in capturing the temporal dependencies

inherent in sign language gestures [2]. Our approach aligns with the findings of Natrajan et al., who also employed LSTM networks in their framework for sign language recognition and reported promising results [7].

Furthermore, the quality and diversity of our training dataset plays a crucial role in achieving high accuracy levels. Our dataset, inspired by the methodologies outlined in the works of T.M Reddy and G. Anantha Rao, encompasses a wide range of sign language gestures captured from various angles and lighting conditions [3, 4]. This diversity enables our model to generalize well to unseen data and adapt to different signing styles, as highlighted by Muntadher Khamees in his real-time American Sign Language recognition system [8].

To enhance the robustness of our model and mitigate overfitting, we employ data augmentation techniques such as random rotations, translations, and flips. These techniques, inspired by the research of Amrita Thakur, have proven effective in improving the generalization capabilities of deep learning models for hand gesture recognition tasks [5].



Fig 7.3 Showing 100% accuracy for the Letter B



Fig. 7.4 Showing 100% accuracy for the Letter A

Additionally, our efficient training methodologies draw inspiration from various optimization strategies outlined in the referenced papers. Adaptive learning rate scheduling and early stopping, as suggested by previous research, contribute to the rapid convergence of our model and prevent overfitting to the training data [1, 3].

In summary, our sign language recognition system achieves impressive accuracy levels, demonstrating the effectiveness of the model architecture, training methodology, and dataset used. By building upon the findings and methodologies proposed in the referenced papers, our study contributes to the ongoing progress in hand gesture recognition and lays the foundation for more robust and efficient interaction paradigms for individuals with hearing impairments.

7. CONCLUSION

In conclusion, the development of a sign language recognition system using deep learning techniques holds significant promise for enhancing communication and accessibility for individuals with hearing impairments. Through the utilization of recurrent neural networks, specifically Long Short-Term Memory (LSTM) networks, coupled with a comprehensive dataset and efficient training methodologies, the system has demonstrated impressive accuracy levels, reaching approximately 99% after just 200 epochs of training.

In the future, we aim to enhance the sign language recognition system by advancing its capabilities beyond single signs to recognize phrases and even complete sentences. This expansion will involve leveraging the contextual information present in sequences of signs, enabling more accurate and natural language understanding. To achieve this, we plan to explore

sequence-to-sequence models, such as encoder-decoder architectures, which have shown promise in various natural language processing tasks. By training the model on annotated datasets containing sign language phrases and sentences, we can teach it to understand the grammatical and syntactical structures inherent in sign language communication. Additionally, we will investigate techniques for incorporating multimodal inputs, such as video and skeletal data, to provide richer context and improve the robustness of the system across different signing styles and environments. Through these advancements, we aim to create a more comprehensive and effective sign language recognition system that better serves the needs of individuals with hearing impairments, promoting greater accessibility and inclusivity in communication.

REFERENCES

- [1] N. K. Bhagat, Y. Vishnusai and G. N. Rathna, "Indian Sign Language Gesture Recognition using Image Processing and Deep Learning," 2019 Digital Image Computing: Techniques and Applications (DICTA), Perth, WA, Australia, 2019, pp. 1-8, doi: 10.1109/DICTA47822.2019.8945850.
- [2] B. Natarajan et al., "Development of an End-to-End Deep Learning Framework for Sign Language Recognition, Translation, and Video Generation," in IEEE Access, vol. 10, pp. 104358-104374, 2022, doi: 10.1109/ACCESS.2022.3210543.
- [3] T. M. Reddy, S. Abhishek, V. V. Kalyan, P. R. Varma and S. Sanapala, "Sign Language Recognition Using OpenCV and Convolutional Neural Networks," 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE), Chennai, India, 2023, pp. 1-6, doi: 10.1109/RMKMATE59243.2023.10369046.
- [4] Rasha Amer Kadhim, Muntadher Khamees, "A Real-Time American Sign Language Recognition System using Convolutional Neural Network for Real Datasets TEM Journal. Volume 9, Issue 3, Pages 937-943, ISSN 2217-8309, DOI: 10.18421/TEM93-14, August 2020.
- [5] G. A. Rao, K. Syamala, P. V. V. Kishore and A. S. C. S. Sastry, "Deep convolutional neural networks for sign language recognition," 2018 Conference on Signal Processing and Communication Engineering Systems (SPACES), Vijayawada, India, 2018, pp. 194-197, doi: 10.1109/SPACES.2018.8316344.
- [6] Thakur, Amrita, et al. "Real time sign language recognition and speech generation." Journal of Innovative Image Processing 2.2 (2020): 65-76.
- [7] Dong, C., Leu, M. C., & Yin, Z. (2015). American sign language alphabet recognition using Microsoft kinect. In Proceedings of the IEEE conference on computer vision and pattern recognition workshops (pp. 44-52).
- [8] Singha, J., & Das, K. (2013). Hand gesture recognition based on Karhunen-Loeve transform. arXiv preprint arXiv:1306.2599.
- [9] Kang, B., Tripathi, S., & Nguyen, T. Q. (2015, November). Real-time sign language fingerspelling recognition using convolutional neural networks from depth map. In 2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR) (pp. 136-140). IEEE.
- [10] C.K.M. Lee, Kam K.H. Ng, Chun-Hsien Chen, H.C.W. Lau, S.Y. Chung and Tiffany Tsoi, "American sign language recognition and training method with recurrent neural network", November 2020.
- [11] Pei Xu, "A real time hand gesture recognition and human-computer interaction system", Proceeding of the Computer Vision and Pattern Recognition, 2017.
- [12] V. Bhavana, G. M. Surya Mouli and G. V. Lakshmi Lokesh, "Hand Gesture Recognition Using Otsu's Method", 2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), pp. 1-4, 2017
- [13] K. O. Rodriguez and G. C. Chavez, "Finger Spelling Recognition from RGB-D Information Using Kernel Descriptor", 2013 XXVI Conference on Graphics Patterns and Images, 2013
- [14] Mukul Singh Kushwah, Manish Sharma, Kunal Jain and Anish Chopra, "Sign language interpretation using pseudo glove", Proceeding of International Conference on Intelligent Communication Control and Devices, pp. 9-18, 2017.
- [15] Yann LeCun, Yoshua Bengio and Geoffrey Hinton, Deep learning. Nature, vol. 521, no. 7553, pp. 436-444, 2015.
- [16] Alex Krizhevsky, Ilya Sutskever and E Hinton Geoffrey, "Imagenet classification with deep convolutional neural networks", Advances in neural information processing systems, pp. 1097-1105, 2012.
- [17] Yann LeCun, Yoshua Bengio and Geoffrey Hinton, Deep learning. Nature, vol. 521, no. 7553, pp. 436-444, 2015.
- [18] Becky Sue Parton, "Sign language recognition and translation: A multidisciplined approach from the field of

- artificial intelligence", Journal of deaf studies and deaf education, vol. 11, no. 1, pp. 94-101, 2005.
- [19] Canzler, U., Dziurzyk, T.: Extraction of non-manual features for videobased sign language recognition. In: Proceedings of the IAPR Workshop on Machine Vision Applications, pp. 318–321. Nara, Japan (2002)
- [20] S. Tamura and S. Kawasaki, Recognition of sign language motion images, In Pattern Recognition, volume 21, pages 343-353, 1988
- [21] J. Ma, W. Gao, C. Wang, and J. Wu, A continuous Chinese Sign Language recognition system, International Conference on Automatic Face and Gesture Recognition, pages 428-433, 2000
- [22] N. Tanibata, N. Shimada, and Y. Shirai, Extraction of hand features for recognition of sign language words, In Proc. Intl Conf. Vision Interface, pages 391-398, 2002.
- [23] Yann LeCun, Yoshua Bengio and Geoffrey Hinton, Deep learning. Nature, vol. 521, no. 7553, pp. 436-444, 2015.
-

