International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307 (Online), Volume-15 Issue-2, May 2025

Sign Language Detection and Recognition using Image Processing for Improved Communication

Nishtha Bhagyawant, Gauri Tamondkar, Sneha Yadav, Shwethashree Kenche, Sunny Sall

Abstract: ***This study presents an advanced deep learning framework for the real-time recognition and translation of Indian Sign Language (ISL). Our approach integrates Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to effectively capture both the spatial and temporal features of ISL gestures. The CNN component extracts rich visual features from the input sign language videos, while the LSTM component models the dynamic temporal patterns inherent in the gesture sequences. We evaluated our system using a comprehensive ISL dataset consisting of 700 fully annotated videos representing 100 spoken language sentences. To assess the effectiveness of our approach, we compared two different model architectures: CNN-LSTM and SVM-LSTM. The CNN-LSTM model achieved a training accuracy of 84%, demonstrating superior performance in capturing both visual and sequential information. In contrast, the SVM-LSTM model achieved a training accuracy of 66%, indicating comparatively lower effectiveness in this context. One of the key challenges faced during the development of the system was overfitting, primarily due to computational constraints and the limited size of the dataset. Nevertheless, through careful tuning of hyperparameters and the use of various optimization strategies, the model exhibited promising results, suggesting its potential for real-world applications. This paper also discusses the data preprocessing techniques employed, including video frame extraction, normalization, and data augmentation, which played a critical role in enhancing model performance. By addressing the complexities of sign language recognition, our work contributes to advancing communication accessibility for individuals relying on ISL, promoting greater inclusivity through technology.***

Keywords: Sign Language (SL), OpenCV, CNN, LSTM, hand gesture, real-time, Deep Learning (DL)

Abbreviations:
LSTM: Long Short-Term Memory
ISL: Indian Sign Language
SL: Sign Language
TSL: Turkish Sign Language
IPO: Input-Process-Output
KNN: K-Nearest Neighbors
DNN: Deep Neural Networks
ASL: American Sign Language
LVSM: Latent Support Vector Machine
CNNs: Convolutional Neural Networks

Manuscript Received on 23 April 2025 | First Revised Manuscript Received on 27 April 2025 | Second Revised Manuscript Received on 04 May 2025 | Manuscript Accepted on 15 May 2025 | Manuscript published on 30 May 2025.

*Correspondence Author(s)

Nishtha Bhagyawant*, Department of Computer Engineering, St. John College of Engineering and Management, Palghar (Maharashtra), India. Email ID: nishthabhagyawant@gmail.com, ORCID ID: 0009-0005-8697-5195

Gauri Tamondkar, Department of Computer Engineering, St. John College of Engineering and Management, Palghar (Maharashtra), India. Email ID: gauritamondkar5693@gmail.com, ORCID ID: 0009-0006-3882-9377

Sneha Yadav, Department of Computer Engineering, St. John College of Engineering and Management, Palghar (Maharashtra), India. Email ID: snehayadav442003@gmail.com, ORCID ID: 0009-0004-8072-2050

Shwethashree Kenche, Department of Computer Engineering, St. John College of Engineering and Management, Palghar (Maharashtra), India. Email ID: shwethashree9873@gmail.com, ORCID ID: 0009-0007-1947-7748

Sunny Sall, Department of Computer Engineering, St. John College of Engineering and Management, Palghar (Maharashtra), India. Email ID: sunnys@sjcem.edu.in, ORCID ID: 0000-0002-8955-4952

I. INTRODUCTION

Effective communication is essential for societal survival. Sign language (SL) serves as the primary mode of communication for individuals who are deaf. However, there are notable variations within sign language, leading to challenges in developing accurate recognition systems. These challenges have resulted in limited attempts to recognize SL gestures effectively, particularly for continuous, sentence-level SL.

Sign language (SL) is a visuospatial language that relies on hand shapes, movements, and facial expressions to convey meaning. Recognizing and interpreting these dynamic gestures is a complex task due to significant variability across individuals, contexts, and even within the same language. Previous research has largely focused on recognizing isolated SL alphabets or words, leaving the recognition of continuous, sentence-level SL as an open challenge.

In this paper, we introduce an innovative approach that combines the strengths of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to address the complexities of continuous Indian Sign Language (ISL) recognition. The CNN module efficiently extracts the spatial characteristics of hand shapes and gestures, while the LSTM component captures the temporal dynamics of sign language sequences. By integrating these complementary neural network architectures, our system learns robust representations of ISL, enabling accurate real-time recognition and translation.

Additionally, we compared the performance of two different model architectures: CNN-LSTM and SVM-LSTM. The CNN-LSTM model achieved an 84% training accuracy, while the SVM-LSTM model achieved a 66% training accuracy. Despite facing challenges with overfitting due to computational constraints, our system demonstrated promising results, paving the way for improved communication accessibility for the deaf community.

II. LITERATURE SURVEY

Previous research on sign language recognition has explored diverse methodologies, such as machine learning, computer vision, and deep learning techniques. Below is a brief overview of the pertinent literature.

A. Leap Motion-Based Myanmar Sign Language Recognition using Machine Learning [1]

A technique for enhancing skin color and a color-based segmentation approach is utilized to identify hand skin tones and manual gestures within a machine learning-based Myanmar Sign Language recognition system. Face identification can detect a face's frontal aspect; however, it occasionally struggles to identify unconstrained faces.

B. Automatic Bengali Sign Language Understanding with a New Deep Convolutional Neural Network [2]

This study provided a dataset containing images of Bengali Sign Language, which was used to apply a CNN architecture, achieving 99.6% accuracy. However, the study does not detail the steps taken to obtain the results or the tools used to train the model.

C. Detection of Turkish Sign Language Using Deep Learning and Image Processing [3]

This study introduces a deep learning model incorporating Capsule Networks (CapsNet) and Telnet systems, designed for recognizing the Turkish Sign Language alphabet to assist individuals with hearing impairments. Only letters belonging to TSL were identified; words, numbers, or expressions were not discerned. The TSL Net model was found to perform less effectively in practical settings.

D. Automatic Recognition of Arabic Alphabets Sign Language using Deep Learning [4]

This paper analyzes the offline classification and recognition performance of three widely used deep learning models (AlexNet, VGGNet, and GoogleNet/Inception) for the Arabic Sign Language alphabet. The most recent Arabic Sign Language dataset (ArSL 2018), consisting of 54,000 images, was used to train and test the models.

E. Automatic Translation of American Sign Language with Deep Learning Cameras [5]

This project aims to develop a vision-based application to enhance communication between signers and non-signers by translating sign language into text. The model extracts spatial and temporal features from video sequences, using a CNN (Inception) for spatial feature extraction and an RNN for temporal dynamics. However, variations in signer facial features led to decreased model accuracy when faces were included. Consequently, the videos were edited to focus only on gestures up to the neck.

F. Sign-Language Recognition using Modified Convolutional Neural Networks [6]

This work employs the I3D Inception model for Sign Language Recognition using a transfer learning approach. Testing showed that 10 words and 10 signs were accurately signed across 100 classes. However, the model's validation accuracy was not very high, and in some cases, it overfitted.

G. Real-Time Recognition of Indian Sign Language [7]

Using the FCM Algorithm, the system achieved a gesture labeling accuracy of 75% and could recognize 40 ISL words in real-time. However, it consumed significant computing time and performed poorly with high-dimensional datasets.

H. Sign Quiz: An ASLR-Based Quiz Training Tool for Fingerspelled Signs in Indian Sign Language [8]

The online tool Sign Quiz, which teaches sign language using Deep Neural Networks (DNN), is presented in this paper. Its detection accuracy threshold was set at 85%. Some handwritten signs were incorrectly interpreted because the letters were not sufficiently large to fill the screen.

I. An Deep Learning-Based Approach for Assisting the Hard of Hearing in Emergencies [9]

This research introduces algorithms for identifying and categorizing hand gestures in emergency sign language (ISL) video data. The classification model, combining pre-trained VGG-16 and LSTM networks, achieved an accuracy of 98%. The detection model, utilizing YOLOv5, reached a mean average precision (mAP) of 99.6%. Together, these models form a system capable of recognizing both static and dynamic hand gestures from video frames. Furthermore, dynamic gestures were identifiable with a smaller dataset.

J. Mudra: Indian Sign Language Translator for Banks using Convolutional Neural Networks [10]

This system aims to translate bank-related sign language patterns into text, enabling deaf-mute individuals to complete banking procedures independently. An accuracy of 81% was achieved when testing the entire dataset. However, the algorithm occasionally confused similar terms, such as "tellers," "working hours," and "balance sheets".

K. Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on Input-Process-Output [11]

This paper presents a comprehensive review of various sign language recognition systems designed for deaf-mute individuals, focusing on the input-process-output (IPO) model. It categorizes existing technologies based on their data acquisition methods, processing techniques, and output modalities. The study highlights that while many systems show promising results, challenges remain in terms of scalability, real-time processing, and accuracy across diverse sign languages and user conditions.

L. Vision-Based Hand Gesture Recognition [12]

This study explores a vision-based approach to recognizing hand gestures using image processing techniques. The system primarily focuses on extracting hand features through edge detection and contour analysis. Although effective in controlled environments, the approach struggles with background noise, varying lighting conditions, and occlusions, limiting its robustness in real-world applications.

M. Saliency-Based Alphabet and Numbers of American Sign Language Recognition Using Linear Feature Extraction [13]

This work proposes a method for recognizing American Sign Language (ASL) alphabets and numbers by using saliency-based detection combined with linear feature extraction. The approach enhances important visual regions before applying classification algorithms, leading to improved recognition accuracy. However, the method's effectiveness is limited when dealing with overlapping gestures or low-resolution inputs.

N. Latent Support Vector Machine Modelling for Sign Language Recognition with Kinect [14]

This study presents a sign language recognition system using Kinect sensor data and latent support vector machine (LSVM) models. By leveraging depth information and skeletal tracking, the method achieves improved gesture classification. However, system accuracy drops with fast hand movements or occlusions.

O. American Sign Language-Based Finger-Spelling Recognition Using K-Nearest Neighbors Classifier [15]

This paper describes an ASL finger-spelling recognition approach using the K-nearest neighbors (KNN) algorithm. The system effectively classifies static hand gestures with high accuracy on small datasets. Nonetheless, its performance declines as the complexity and volume of input gestures increase.

P. Deep Learning in Vision-Based Static Hand Gesture Recognition [16]

This work explores the use of deep learning, particularly convolutional neural networks (CNNs), for static hand gesture recognition. The proposed method achieves strong accuracy under controlled conditions, but its robustness decreases in cluttered or dynamic environments.

Q. Artificial Neural Network-Based Method for Indian Sign Language Recognition [17]

The authors propose an artificial neural network (ANN) approach for recognizing Indian Sign Language (ISL) gestures. The system demonstrates reasonable recognition rates for isolated signs but faces limitations when attempting continuous gesture recognition due to contextual variability.

R. Multimodal Machine Learning for Sign Language Prediction [18]

This recent study introduces a multimodal machine learning framework combining visual, motion, and contextual inputs to improve sign language prediction. The integration of multiple data types enhances recognition accuracy, although challenges remain in synchronizing heterogeneous input streams.

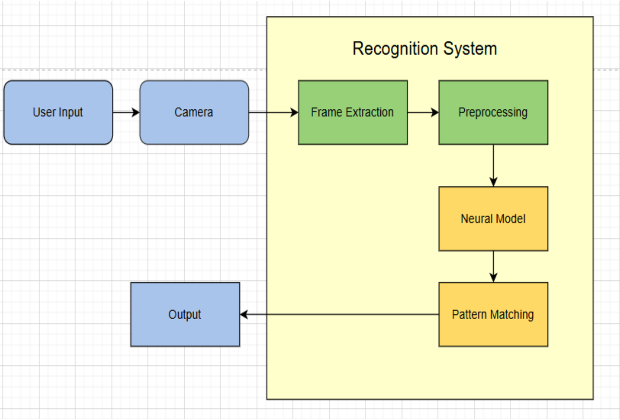
S. Real-Time Computer Vision-Based Bengali Sign Language Recognition [19]

This paper presents a real-time system for recognizing Bengali Sign Language (BdSL) using computer vision techniques. It demonstrates effective recognition for a limited vocabulary set but struggles with scalability to larger vocabularies and varying signer styles.

T. ISL-CSLTR: Indian Sign Language Dataset for Continuous Sign Language Translation and Recognition [20]

This work introduces the ISL-CSLTR dataset, designed for continuous sign language translation and recognition in Indian Sign Language [21]. The dataset significantly contributes to the research community but presents challenges related to signer variability and long continuous gesture sequences [22].

III. PROPOSED METHODOLOGY



[Fig.1: Block Diagram of Proposed System]

The proposed system for recognizing and translating Indian Sign Language (ISL) [23] involves a comprehensive recognition system that processes user input through a camera and utilizes multiple steps to produce an output [24]. The system follows a structured process [25]:

- **User Input:** The initial input is provided by the user through sign language gestures.
- **Camera Capture: A** camera captures these gestures and converts them into digital frames for processing.
- **Recognition System:**
- **Frame Extraction:** Frames are extracted from the captured input for further analysis.
- **Preprocessing:** These frames undergo preprocessing to enhance their quality and ensure they are suitable for analysis.
- **Neural Model:** The pre-processed frames are input into the neural model, specifically a CNN-LSTM model.
- **Pattern Matching:** The neural model's output is used to identify and match patterns corresponding to sign language gestures.
- **Output Generation:** The final recognized output is generated and presented.

A. CNN & LSTM Model

- **Fully Convolutional Layers:** These layers apply convolution operations to the input frames, extracting spatial features such as edges and shapes.
- **Pooling Layers:** These layers reduce the dimensions of the feature maps, retaining essential information while lowering computational load.
- **Activation Layers:** Non-linear activation functions (e.g., ReLU) are applied to introduce non-linearity into the model, enabling it to learn complex patterns.
- **Connected Layers:** These layers integrate features extracted by the convolutional layers, forming a high-level representation of the input frames.

B. Long Short-Term Memory (LSTM) Layers

- **LSTM Units:** These units process sequential data from the CNN layers, capturing temporal dependencies and

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

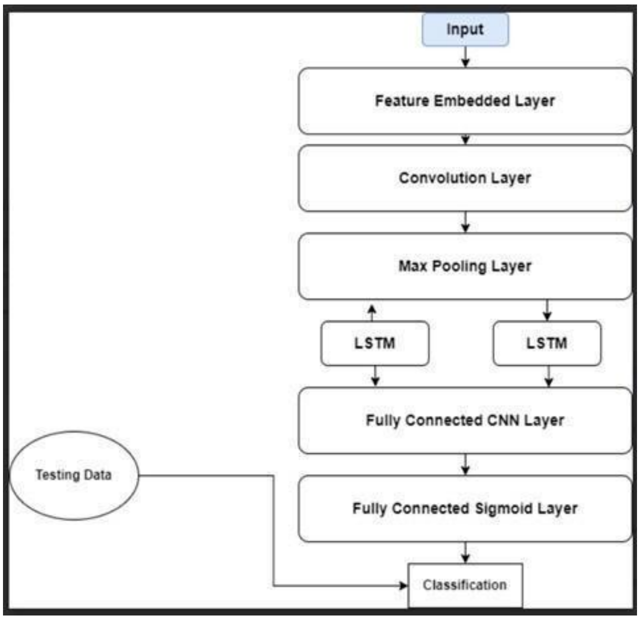
patterns over time. LSTMs are effective in retaining information across long sequences, making them suitable for analyzing sequences of frames.

- **Output Layer:** The final output from the LSTM layers is used for pattern matching and recognition.

By combining the strengths of CNN and LSTM layers, the model effectively processes both spatial and temporal features, leading to accurate recognition of sign language gestures.

IV. IMPLEMENTATION

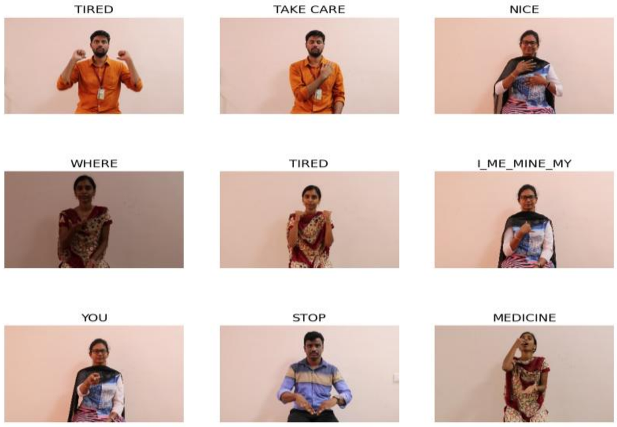
In this setup, the user inputs real-time data through a camera, which is then preprocessed using Media Pipe and OpenCV. Following preprocessing, key points are extracted. These key points and motion patterns are processed by the CNN and LSTM models to predict real-time sign language gestures. The model architecture includes Conv2D, Max Pooling, LSTM, and Dense layers. Various techniques were employed to manage the complexity of the model. Each layer was configured with 32 neurons, and the image input shape was set to 128 × 128 × 3 (height × width × number of channels). The model was trained for 15 epochs with a batch size of 32.



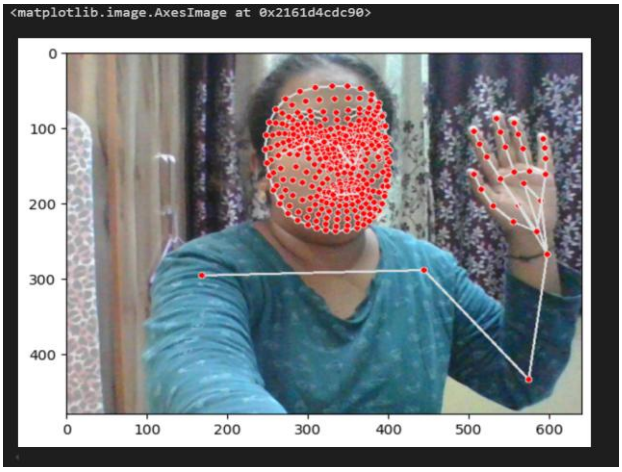
[Fig.2: CNN-LSTM Model]

A. Training and testing dataset

The project for the translation and recognition of sign language uses a fully labeled, sentence-level Indian Sign Language dataset. “With 700 fully annotated films, 18,863 sentence-level frames, and 1,036 word-level images for 100 spoken language sentences delivered by seven distinct signers, the ISL-Corpus has a sizable vocabulary”. This freely available corpus is organized according to signer variants and temporal boundaries and comes with comprehensive annotations. The dataset will be split into two groups: 80% for training and 20% for testing. It consists of both images and videos of the sentences.



[Fig.3: Dataset Images After Preprocessing Using Open CV]



[Fig.4: Image of Extract key Points from Frames to Preprocess for Recognition Using Media Pipe]

B. Hardware and software requirements

The hardware and software utilized in this project are outlined in the table below.

Table-I: Hardware Requirements Used for the Project.

Properties	Requirement
Processor	2.4 GHz, intel i3 or i5
RAM	4 GB
Free Storage	16 GB

Table-II: Software Requirement for the Project

Properties	Requirement
Framework	Media Pipe, OpenCV
Operating System	Linux, Windows
Tools	Visual Studio
Language	Python 3.8.10

C. Preprocessing techniques

Data preprocessing is a crucial initial phase in data mining. It involves refining raw data through cleaning and transformation to render it suitable for analytical purposes. As shown in Fig. 3, this work outlines the steps involved in cleaning, transforming, and merging data to prepare it for analysis. This process aims to enhance data quality and tailor it to specific data mining tasks. The resolution of the images present in the dataset is 1920 × 1080 in JPG format, and the video dataset is in MP4 format.

i. Image Resizing

Before supplying the input photos to the CNN-LSTM model, they are resized to a

Retrieval Number: 100.1/ijsce.B366815020525
DOI: 10.35940/ijsce.B3668.15020525
Journal Website: www.ijsce.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

standard size. This ensures uniform image dimensions, which is crucial for the input layer of the model.

ii. Frame Selection

For video sequences, keyframes that best represent the pertinent information are selected. This simplifies computation and focuses on key moments.

iii. Frame Difference (For Video Sequence)

The difference between two consecutive frames is determined to obtain motion information, which is essential for recognizing sign language.

iv. Background Subtraction

Backgrounds are removed to accentuate the hand gestures in the foreground.

v. Frame Normalization

Pixel values in every frame of a video clip are normalized.

D. Libraries

A collection of similar modules is referred to as a Python library. It includes code packages that are frequently utilized in various applications, simplifying and facilitating Python programming for developers.

i. OpenCV

OpenCV is an open-source library initially created by Intel and now maintained by a global community of developers. In this research, OpenCV was utilized for image processing tasks. Its versatile libraries enabled efficient preprocessing of ISL video frames, including operations such as resizing, normalization, and noise reduction. These preprocessing steps enhanced the quality of the frames, ensuring that subsequent feature extraction and recognition processes were more effective and accurate.

ii. TensorFlow

TensorFlow is an open-source platform specifically designed for building comprehensive artificial intelligence applications. As shown in Fig. 4, MediaPipe Holistic was utilized to extract features from Indian Sign Language (ISL) video sequences. MediaPipe Holistic combines multiple models to capture key points from the hands, face, and body, providing a comprehensive understanding of gestures. These extracted features were fed into the neural network models, enhancing the accuracy and robustness of ISL recognition.

E. Training Process and Optimization Techniques

To mitigate overfitting and enhance model performance, various techniques were employed. As previously mentioned, the dataset size was 8 GB, containing numerous complex patterns. To boost the model's performance, data augmentation strategies were implemented to enable efficient training despite limited data availability.

To address the challenges of overfitting and enhance the model’s generalization capabilities, the following optimization techniques were employed:

i. Data Augmentation

Various data augmentation methods were utilized to increase the diversity of the training dataset.

ii. Adaptive Optimization

The Adam optimizer was employed with a learning rate schedule, starting at 0.001 and decreasing by a factor of 0.95 after the initial 10 epochs.

iii. Mixed Precision Training

To improve computational efficiency and prevent kernel crashes, mixed precision training was utilized, leveraging both float16 and float32 data types.

iv. Early Stopping

An early stopping criterion was applied to monitor the validation loss and mitigate overfitting.

V. ANALYSIS AND RESULT

To evaluate our proposed system for Indian Sign Language (ISL) recognition, we implemented and compared two model architectures: CNN-LSTM and SVM-LSTM. Both models were trained and tested on the same dataset to ensure a fair comparison.

A. Challenges and Computational Constraints

The primary limitation was the impact of computational constraints on model training and performance. The large size and complexity of the ISL-Corpus dataset, combined with the high computational demands of the CNN-LSTM architecture, led to overfitting and suboptimal model convergence. Despite employing optimization techniques, limited resources during training compromised the model’s effectiveness. Increasing the input size and model complexity resulted in kernel crashes, highlighting the need for more powerful computational resources.

B. CNN-LSTM Model

The CNN-LSTM model combines Convolutional Neural Networks (CNNs) for spatial feature extraction with Long Short-Term Memory (LSTM) networks for capturing temporal patterns. This model achieved a training accuracy of 84% but faced challenges with overfitting, as evidenced by a validation loss of 5.57 and a validation accuracy of 100%. The model tended to memorize noise rather than learning meaningful patterns due to computational constraints.

C. SVM-LSTM Model

The SVM-LSTM model integrates Support Vector Machines (SVMs) with LSTM networks. SVMs handle feature extraction, while LSTMs capture temporal dynamics. This model achieved a lower training accuracy of 66%. The SVM-LSTM model’s reduced performance can be attributed to its less effective feature extraction capabilities.

D. Comparison and Results

The CNN-LSTM model outperformed the SVM-LSTM model in ISL recognition, demonstrating its superiority in capturing and analyzing both spatial and temporal features. The CNN-LSTM model shows significant promise in enhancing communication accessibility for the deaf community through the precise, real-time recognition and translation of Indian Sign Language.

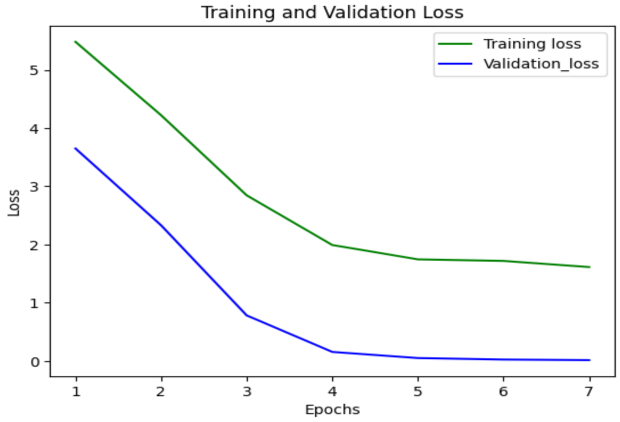
```
multilabel_confusion_matrix(ytrue, yhat)

array([[[4, 1],
        [0, 0]],

       [[3, 0],
        [0, 2]],

       [[2, 0],
        [1, 2]]], dtype=int64)
```

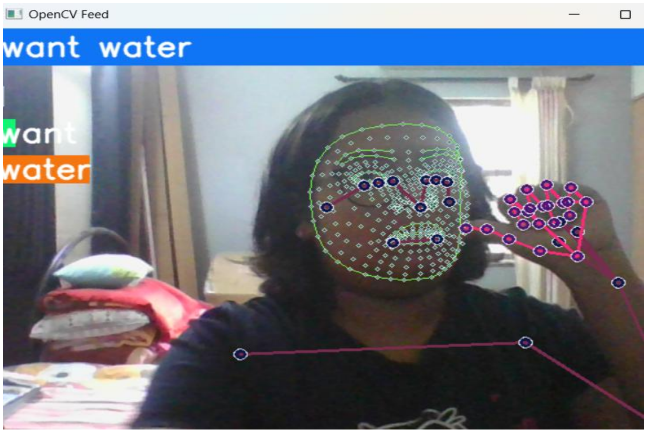
[Fig.5: CNN-LSTM Model's Confusion Matrix]



[Fig.6: Graph of Training and Validation Loss]

```
Average Train Loss: 3.355898118019104
Average Train Accuracy: 0.8428650056773966
Average Validation Loss: 5.5705864754272625
Average Validation Accuracy: 1.0
Average Test Loss: 5.5705864754272625
Average Test Accuracy: 1.0
```

[Fig.7: Accuracy Obtained after Training the Model]



[Fig.8: Image of Real-Time Working of the CNN-LSTM Model on Video Input]

VI. FUTURE WORK

Looking ahead, the focus is on overcoming computational barriers by exploring more efficient model architectures and utilizing advanced hardware such as high-performance GPUs. We also plan to enhance the system's ability to manage linguistic variations, regional dialects, and a broader range of sign language expressions, making it more accessible and inclusive for the deaf-mute community. Future plans include incorporating additional languages and integrating text-to-speech and sign-to-text functionalities. We aim to create a precise and comprehensive dataset for Indian Sign Language, which remains under-researched. As technology advances and more languages are included, the system's capacity to assist the deaf and mute community is expected to grow.

VII. CONCLUSION

This research introduces an innovative CNN-LSTM framework for the real-time detection and translation of Indian Sign Language. By leveraging the advantages of convolutional neural networks and recurrent neural networks, the proposed system can accurately capture the spatial and temporal characteristics of sign language gestures.

Despite the challenges faced during the training process, the model demonstrated promising results on the ISL-Corpus dataset, achieving an accuracy of 84%. The extensive analysis and discussion of the model's performance, limitations, and future research directions provide valuable insights for advancing sign language recognition technology.

This model aims to enhance communication accessibility for individuals with hearing or speech impairments and those not proficient in sign language. To achieve this, the dataset includes a variety of commonly used daily sentences along with a separate folder containing essential symbols.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/ Competing Interests:** Based on my understanding, this article has no conflicts of interest.
- **Funding Support:** This article has not been funded by any organizations or agencies. This independence ensures that the research is conducted with objectivity and without any external influence.
- **Ethical Approval and Consent to Participate:** The content of this article does not necessitate ethical approval or consent to participate with supporting documentation.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Authors Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Z. Hein, T. P. Htoo, B. Aye, S. M. Htet and K. Z. Ye, "Leap Motion based Myanmar Sign Language Recognition using Machine Learning", 2021 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), St. Petersburg, Moscow, Russia, 2021, pp. 2304-2310, DOI: http://doi.org/10.1109/ElConRus51938.2021.9396496
2. M. J. Hossein and M. Sabbir Ejaz, "Recognition of Bengali Sign Language using Novel Deep Convolutional Neural Network", 2020 2nd International Conference on Sustainable Technologies for Industry 4.0

(STI), Dhaka, Bangladesh, 2020, pp. 1-5, DOI: http://doi.org/10.1109/STI50764.2020.9350418
3. Bekir Aksoy, Osamah Khaled Musleh Salman, Özge Ekrem, "Detection of Turkish Sign Language Using Deep Learning and Image Processing Methods", Taylor & Francis, Applied Artificial Intelligence 2021, VOL. 35, NO. 12, 952–981, DOI: https://doi.org/10.1080/08839514.2021.1982184.
4. Rehab Mustafa Duwairi, Zain Abdullah Halloush, "Automatic recognition of Arabic alphabets sign language using deep learning", International Journal of Electrical and Computer Engineering (IJECE), Vol. 12, No. 3, June 2022, pp. 2996~3004, ISSN: 2088-8708, DOI: http://doi.org/10.11591/ijece.v12i3.pp2996-3004.
5. K. Bantupalli and Y. Xie, "American Sign Language Recognition using Deep Learning and Computer Vision", 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 4896-4899, DOI: http://doi.org/10.1109/BigData.2018.8622141
6. Suharjito, H. Gunawan, N. Thiracitta and A. Nugroho, "Sign Language Recognition Using Modified Convolutional Neural Network Model", 2018 Indonesian Association for Pattern Recognition International Conference (INAPR), Jakarta, Indonesia, 2018, pp. 1-5, DOI: http://doi.org/10.1109/INAPR.2018.8627014
7. H. Muthu Mariappan and V. Gomathi, "Real-Time Recognition of Indian Sign Language", 2019 International Conference on Computational Intelligence in Data Science (ICCIDS), Chennai, India, 2019, pp. 1-6, DOI: http://doi.org/10.1109/ICCIDS.2019.8862125
8. J. Joy, K. Balakrishnan and M. Sreeraj, "SignQuiz: A Quiz Based Tool for Learning Fingerspelled Signs in Indian Sign Language Using ASLR", in IEEE Access, vol. 7, pp. 28363-28371, 2019, DOI: http://doi.org/10.1109/ACCESS.2019.2901863
9. Q. M. Areeb, Maryam, M. Nadeem, R. Alroobaea and F. Anwer, "Helping Hearing-Impaired in Emergency Situations: A Deep Learning-Based Approach", in IEEE Access, vol. 10, pp. 8502-8517, 2022, DOI: http://doi.org/10.1109/ACCESS.2022.3142918
10. G. Jayadeep, N. V. Vishnupriya, V. Venugopal, S. Vishnu and M. Geetha, "Mudra: Convolutional Neural Network based Indian Sign Language Translator for Banks", 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), Madurai, India, 2020, pp. 1228-1232, DOI: http://doi.org/10.1109/ICICCS48265.2020.9121144
11. Anderson, R., Wiryana, F., Ariesta, M. C. & Kusuma, G. P., "Sign language recognition application systems for deaf-mute people: A review based on input-process-output", Procedia Comput. Sci. 116, 441–448, DOI: https://doi.org/10.1016/j.procs.2017.10.028.
12. Garg P, Aggarwal N, Sofat S (2009), "Vision based hand gesture recognition". World Acad Sci Eng Technol 49(1):972–977, DOI: http://doi.org/10.5281/zenodo.1074855
13. Zamani M, Kanan HR (2014), "Saliency based alphabet and numbers of American Sign Language recognition using linear feature extraction". In: 4th IEEE International eConference on computer and knowledge engineering (ICCKE), pp 398–403, DOI: http://doi.org/10.1109/ICCKE.2014.6993442
14. Sun C, Zhang T, Xu C (2015), "Latent support vector machine modeling for sign language recognition with Kinect". ACM Trans Intell Syst Technol: TIST 6(2):20, DOI: http://doi.org/10.1145/2629481
15. Aryanie D, Heryadi Y (2015), "American Sign Language-based finger-spelling recognition using k-nearest neighbors' classifier". In: 3rd IEEE international conference on information and communication technology (ICoICT), pp 533–536, DOI: http://doi.org/10.1109/ICoICT.2015.7231481
16. Oyedotun OK, Khashman A (2017), "Deep learning in vision-based static hand gesture recognition". Neural Comput Appl 28(12):3941–3951, DOI: http://doi.org/10.1007/s00521-016-2294-8
17. Adithya V, Vinod PR, Gopalakrishnan U (2013), "Artificial neural network based method for Indian Sign Language recognition". In: IEEE conference on information & communication technologies (ICT), pp 1080–1085, DOI: http://doi.org/10.1109/CICT.2013.6558259
18. Khalafaoui, Y., Grozavu, N., Matei, B., Rogovschi, N. (2024). "Multimodal Machine Learning for Sign Language Prediction". In: Sontea, V., Tiginyanu, I., Railean, S. (eds) 6th International Conference on Nanotechnologies and Biomedical Engineering. ICNBME 2023. IFMBE Proceedings, vol 92. Springer, Cham. DOI: https://doi.org/10.1007/978-3-031-42782-4_26
19. Rahaman MA, Jasim M, Ali MH, Hasanuzzaman M (2014), "Real-time computer vision-based Bengali Sign Language recognition". In: 17th IEEE international conference on computer and information technology (ICCIT), pp 192–197, DOI: http://doi.org/10.1109/ICCITechn.2014.7073150
20. R, Elakkiya; B, NATARAJAN (2021), "ISL-CSLTR: Indian Sign Language Dataset for Continuous Sign Language Translation and Recognition", Mendeley Data, V1, DOI: http://doi.org/10.17632/kcmpdxky7p.1
21. Rajesh B. Mapari, Govind Kharat. (2016). Performance Analysis of Different Classifiers for American Sign Language Recognition. International Journal of Soft Computing and Engineering. (Vol. 6, Issue 1, pp. 90–95). https://www.ijsce.org/wp-content/uploads/papers/v6i1/A2789036116.pdf
22. Singh, S., Gupta, A. K., & Singh, T. (2019). Sign Language Recognition using Hybrid Neural Networks. In International Journal of Innovative Technology and Exploring Engineering (Vol. 9, Issue 2, pp. 1092–1098). DOI: https://doi.org/10.35940/ijitee.l3349.129219
23. Barde, U., & Ghotkar, A. (2020). Sign Language Recognition-A Survey of Techniques. In International Journal of Recent Technology and Engineering (IJRTE) (Vol. 9, Issue 2, pp. 854–857). DOI: https://doi.org/10.35940/ijrte.b3925.079220
24. R, Renjitha., D, Rubashree., V, Priya., & C, Preetha. (2020). Generating an Audio and Text for Indian Sign Language. In International Journal of Engineering and Advanced Technology (Vol. 9, Issue 3, pp. 4371–4374). DOI: https://doi.org/10.35940/ijeat.c6542.029320
25. M R, Dr. P. (2022). Sign Language Recognition System. In Indian Journal of Software Engineering and Project Management (Vol. 2, Issue 1, pp. 1–3). DOI: https://doi.org/10.54105/ijsepm.c9011.011322

AUTHOR'S PROFILE

Nishtha Bhagyawant, completed her Bachelor of Engineering in Computer Engineering from St. John College of Engineering and Management, affiliated with the University of Mumbai. She also holds an Honors degree in Blockchain technology. Nishtha has a strong interest in Data Science, Analytics, and Machine Learning, areas in which she has completed multiple internships and worked on various real-world projects. These experiences have helped her build practical skills, gain valuable industry knowledge, and strengthen her technical foundation. She is passionate about leveraging data-driven solutions to solve complex problems and continues to expand her expertise through continuous learning, certifications, and active participation in workshops and seminars.

Gauri Tamondkar is a Computer Engineering graduate who earned her Bachelor of Engineering (BE) degree with a focus on Data Science and Java programming. Her strong academic background, combined with her passion for these fields, positions her well for a successful career in leveraging data-driven insights to address complex challenges. Gauri has developed technical proficiency through coursework and projects, building a solid foundation for future professional opportunities. Outside of academics, she enjoys pencil sketching, which highlights her creativity, patience, and keen attention to detail. This blend of technical expertise and artistic ability makes her a versatile and valuable asset to any innovative and dynamic team.

Sneha Yadav is a Computer Engineering student currently pursuing her Bachelor of Engineering (BE) degree with a concentration in Python programming. She actively participates in discussions, workshops, and projects that enhance her technical knowledge and practical skills. Beyond academics, Sneha has a well-rounded personality and enjoys engaging in activities such as playing sports, exploring nature, and spending quality time with friends. Her dedication, passion for learning, and continuous drive for self-improvement position her well for a successful career in the field of computer engineering. With her balanced approach to academics and personal growth, she is poised to make a positive impact in the technology industry.

Shwethashree Kenche, completed her Bachelor of Engineering in the Computer Engineering Department at St. John College of Engineering and Management, affiliated with the University of Mumbai. She also earned an Honors degree in Data Science. During her academic journey, Shwethashree completed an internship in Data Science and Machine Learning, where she gained hands-on experience working on real-world projects. Her involvement in various projects has significantly enhanced her technical skills and deepened her understanding of Data Science and Analytics. With her strong academic background, practical exposure, and continuous drive to learn, she is well-prepared to build a successful career in the field of data-driven technologies and innovation.

Sunny Sall, holds a Master's degree in Computer Engineering and a Ph.D. in Information Technology from the University of Mumbai, India. He is currently serving as an Assistant Professor and Head of the Department of Data Science at St. John College of Engineering and Management, Palghar (affiliated with the University of Mumbai). His research focuses on addressing challenges and developing solutions in Wireless Sensor Networks. Dr. Sall also has extensive experience in Computer Science, particularly in areas such as System Programming, Compiler Design, Microprocessors, and Assembly Language Programming. His academic and research contributions, along with his leadership, make him a valuable asset in advancing education and innovation in the field.

Retrieval Number: 100.1/ijsce.B366815020525
DOI: 10.35940/ijsce.B3668.15020525
Journal Website: www.ijsce.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)