JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Sign Language Translator

[1] Prof. Jagruti Nagaonkar, [2] Savit Nair, [3]Akhil Mani, [4]Reuben Dsa, [5]Erica Mathias
[1] Assistant Professor,[2] Student ,[3] Student, [4] Student, [5]Student
Department of Computer Engineering, Fr. Conceicao Rodrigues College of Engineering,Mumbai,India

***Abstract*:** This research paper introduces the utilization of frames and various technologies to implement an LSTM deep model for motion-based data analysis. The focus is on analyzing user motion captured through live video streams and training the program using self-assessed live datasets and pre-downloaded sets. The proposed approach leverages the capabilities of LSTM models to capture temporal dependencies in motion data, enabling accurate analysis and prediction of user actions or gestures.

The methodology involves pre-processing the input data by extracting frames and enhancing the quality of captured motion. The LSTM deep model is then trained using a combination of self-assessed datasets and pre-downloaded sets to ensure a diverse range of motion variations. The experiments demonstrate the effectiveness of the LSTM-based approach in accurately analyzing motion data in real-time video streams. The findings highlight the significance of LSTM deep models in capturing temporal dependencies, leading to improved motion-based data analysis and gesture recognition.

Index terms: **Sign language, Detection of sign language, LSTM model**

I. INTRODUCTION

According to recent statistics, India has a limited number of certified sign language interpreters, approximately 250, serving a population between 1.8 million to 7 million individuals with deafness or speech impairments. This results in a high demand for always available interpreting services. Many businesses worldwide face difficulties in providing services to individuals who use sign language. To address this need, our project offers hand movement, finger movement, and facial gesture recognition. We trained the model using sign language videos to enable accurate prediction and translation. The project translates sign language into simple and understandable English text in real-time, providing a user-friendly interface with text output for sign gesture input. The project is a practical and convenient option for understanding sign language, using only cameras to provide the same services as a sign language interpreter. With technological advancements, communication and comprehension have become easier, even for people with disabilities. The project eliminates the need for an intermediary translator, which is often expensive and unavailable. This tool will enhance communication and interaction between individuals who use sign language and the wider community, including various businesses and companies. Additionally, it will increase the employment opportunities for speech-impaired individuals.

Our project has the potential to make a significant impact in the lives of individuals who rely on sign language to communicate. The lack of available sign language interpreters has long been a challenge for the deaf and speech-impaired communities. It often leads to social isolation and exclusion from daily activities that require communication, such as going to the doctor, attending school or university, or accessing essential services. Our project provides a cost-effective and accessible solution that allows individuals to communicate effectively and without the need for a professional interpreter. This technology can also help companies and businesses become more inclusive and accommodating to people with disabilities. It will enable them to provide better services and connect with a wider audience, leading to a positive impact on their business and social responsibility reputation. Moreover, our project has the potential to reduce the language barrier and bridge the gap between different cultures. Sign language is a unique form of communication that transcends spoken language and can be used by people from different countries and backgrounds. The ability to translate sign language in real-time can facilitate communication and understanding between people from different parts of the world, leading to increased social cohesion and integration. In conclusion, our project has the potential to change the lives of millions of individuals with deafness or speech impairments worldwide, as well as the businesses and companies that serve them. It is a step towards creating a more inclusive and accessible society for all.

II. LITERATURE REVIEW

Sign language recognition and translation using machine learning has been an active area of research in recent years. A number of different approaches have been proposed and tested, as reflected in the following research papers.

Anand and Singh proposed a sign language recognition system based on TensorFlow Object Detection API in [1]. They used multi-stream hidden Markov models (MSHMMs), neural networks, Naïve Bayes Classifier, and Support Vector Machine for

processing, achieving accuracy scores ranging from 83.6 to 100. They also used SSD MobileNet v2 object detection model with FPN-lite feature extractor, shared field predictor, and defocus using educational photos scaled to 320x320 for real-time detection. In [2], Abhinav et al. presented a sign language translator using machine learning. They captured an image using a webcam and processed it with Tf pose estimation rules set to identify key elements of the subject's frame, and used a decision tree algorithm to predict gestures based on the resulting data set. Naveen Kumar et al. proposed a real-time sign language detection system using TensorFlow, OpenCV, and Python in [3]. They used transfer learning and a pre-trained SSD mobile internet v2 model to train the system. Prabhu and Muthuselvam developed a BIM sign language translator using machine learning in [4]. They used the Waterfall Model for software development and TensorFlow, Python, OpenCV, and Qt as core development libraries. Overall, these studies demonstrate the effectiveness of machine learning techniques for sign language recognition and translation, and the potential for real-world applications in assisting deaf or hard-of-hearing individuals.
In addition to the studies described above, there have been several other notable efforts to develop sign language recognition and translation systems using machine learning. One such example is the work of Jindal et al. [5], who proposed a sign language recognition system using convolutional neural networks (CNNs) and the LeNet architecture. Their system achieved an accuracy of 93.7% on a dataset of 4000 sign language images. Another study by Ullah et al. [6] proposed a sign language recognition system using a combination of Convolutional Neural Networks and Recurrent Neural Networks (RNNs). They used a dataset of over 6000 images of American Sign Language (ASL) signs and achieved an accuracy of 95.44%. In [7], Bao et al. proposed a real-time American Sign Language (ASL) recognition system using deep learning. They used a combination of convolutional and recurrent neural networks and achieved a recognition accuracy of 99.4% on a test set of 4500 ASL sign images. Li et al. [8] proposed a real-time Chinese sign language recognition system using deep learning. They used a dataset of over 3000 sign language images and achieved an accuracy of 95.8% using a convolutional neural network. Lastly, Huang et al. [9] proposed a sign language translation system that is capable of translating English text to American Sign Language (ASL) video. They used a combination of deep learning techniques, including CNNs and RNNs, to generate ASL video sequences from input English text. Their system achieved an accuracy of 89.4%.

Overall, these studies demonstrate the ongoing development and improvement of sign language recognition and translation systems using machine learning techniques. These systems have the potential to greatly improve communication and accessibility for deaf and hard-of-hearing individuals in a variety of settings.

.

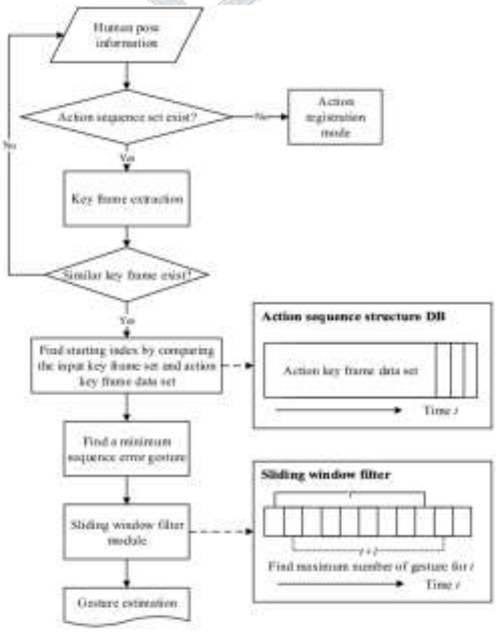
III. DESCRIPTION OF THE SYSTEM OR RESEARCH

- To utilize the frames and various technologies to implement LSTM deep model for analysis of data based on motion of the user through live video stream or to train the program through various datasets (which include self assessed live datasets or pre downloaded sets).

Objective:

- To successfully capture the live camera feed
- To use external libraries and plot the graph on the live camera feed
- To capture specific landmarks through the live camera feed
- To save the last frame of the feed for training purposes
- To train the model using different datasets
- To train the algorithm to suggest respective English words corresponding to the sign language actions.

Proposed System flow:



IV. IMPLEMENTATION DETAILS AND RESULT

A- To distinguish the face and hand gestures.

The algorithm used to identify the face and hand gestures is shown in Fig. 2 and Fig. 3. Using the Open Source Computer Vision (OpenCV) package, Mediapipe, Skicit-learn, Numpy, Tensorflow-Keras, Matplot and other dependencies the image is loaded, scanned, and the frames are extracted. OpenCV recognizes objects, people, and even human faces in photos and videos.

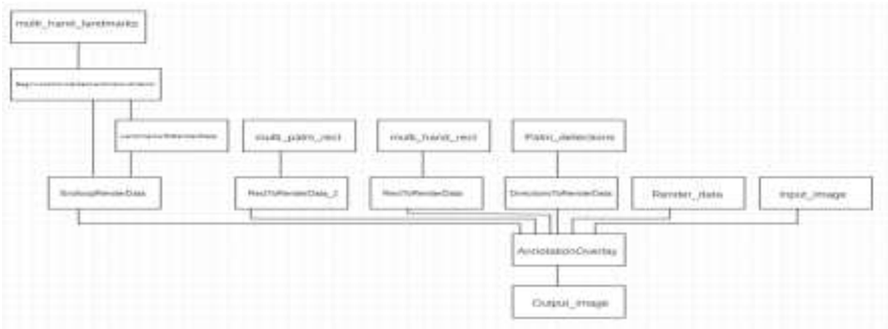


Fig. 2. Identification of hand gestures

The above-mentioned flowchart depicts the flowchart for how exactly the Mediapipe dependency works for hand gesture detection.

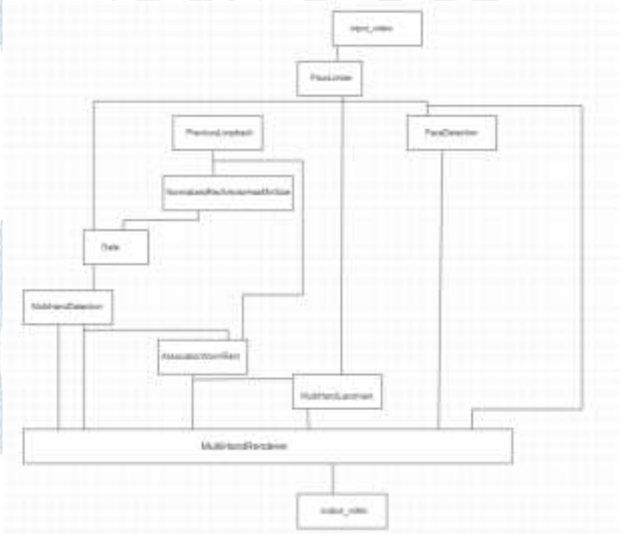


Fig. 3. Identification of Face

Fig. 3. shows an algorithm for detecting a person's face based on the live camera feed of the machine. This module, the Mediapipe dependency, uses OpenCV to convert the given image from RGB to BGR color scheme and ensures that the image is writeable.

LSTM, an artificial neural network used in the fields of artificial intelligence and deep learning, is used to predict the next set of hand gestures and to form the sentences. The LSTM model works by processing each frame of the video input one at a time, and then using the output from each frame to inform the processing of the subsequent frames via probability.

At each time step, the LSTM cell takes in two inputs: the current input frame and the previous hidden state. It then performs several computations, including gating functions that decide which information to keep or discard, and outputs a new hidden state that is passed on to the next time step. The output at each time step can be used to make a prediction about the sign language word or phrase being signed

The poses are predicted by the LSTM model, which learns by training data from the dataset. Fig. 4. illustrates a sample out for this task.

```
pose

[array([ 0.57896292,  0.4660818 , -0.50708061,  0.99954736]),
 array([ 0.60570514,  0.42152685, -0.45998883,  0.99917537]),
 array([ 0.61454785,  0.42483997, -0.46010309,  0.99911642]),
 array([ 0.62237501,  0.42830741, -0.46027249,  0.99909133]),
 array([ 0.56826997,  0.40652329, -0.50364077,  0.9994002 ]),
 array([ 0.55216616,  0.39872205, -0.50349945,  0.99947006]),
 array([ 0.5352782 ,  0.39141929, -0.50397092,  0.99950689]),
 array([ 0.61394292,  0.44444785, -0.13870788,  0.99885565]),
 array([ 0.50065821,  0.396189  , -0.34612101,  0.99964577]),
 array([ 0.57494265,  0.52724564, -0.38038403,  0.99951965]),
 array([ 0.53584433,  0.50849617, -0.43565732,  0.99976248]),
 array([0.62849396, 0.68995136, 0.07486014, 0.99854416]),
 array([ 0.32273027,  0.63861001, -0.32211149,  0.99936849]),
 array([0.7258212 , 0.99631381, 0.09729592, 0.45590517]),
 array([ 0.31704023,  0.98302335, -0.39309856,  0.6924479 ]),
 array([ 0.69210255,  1.15012741, -0.21318731,  0.26684836]),
 array([ 0.58571112,  1.25871909, -0.52954596,  0.49767542]),
 array([ 0.68737185,  1.2043457 , -0.31115481,  0.24643075]),
 array([ 0.63297951,  1.34586072, -0.61219603,  0.47858259]),
 array([ 0.66200358,  1.17448378, -0.32444203,  0.3086839 ]),
 array([ 0.64781821,  1.3168937 , -0.64137471,  0.509592  ]),
 array([ 0.65585011,  1.15919769, -0.23870912,  0.31098557]),
 array([ 0.63053334,  1.28572655, -0.54191494,  0.50562215]),
 array([0.57692033, 1.37362063, 0.05028614, 0.00388429]),
 array([ 0.34748814,  1.37256956, -0.04721932,  0.00419866]),
...
 array([3.62527072e-01, 2.53173923e+00, 3.94251347e-02, 5.31007645e-05]),
 array([5.68262219e-01, 2.61234927e+00, 4.20193106e-01, 1.50102540e-04]),
 array([3.43019873e-01, 2.61064935e+00, 4.02216837e-02, 1.03859988e-04]),
 array([ 5.83743632e-01,  2.70490193e+00, -3.28161120e-02,  2.42164519e-04]),
 array([ 4.23393816e-01,  2.69203734e+00, -5.02568722e-01,  1.27585503e-04])]
```

Fig. 4. Output of Pose detection

B- To scan and predict the gestures

Mediapipe Holistics is employed to detect the positions of gestures. The detected coordinates are then transformed into landmarks that include facial, positional, and hand landmarks. The collected data is stored as arrays using Numpy and categorized into four arrays based on the type of landmarks: pose, face, left_hand, and right_hand. In the event that certain data is not captured in the input feed, a zero-array is created with the desired shape and is stored in the corresponding landmark. A dataset is then created with a designated data path and keywords assigned as subfolders. For each sequence, 30 frames are taken, resulting in 30 folders for each keyword, with each folder containing 30 Numpy arrays representing the 30 sequences for each word.

Fig. 5. demonstrates a sample output for gesture recognition from an input image.

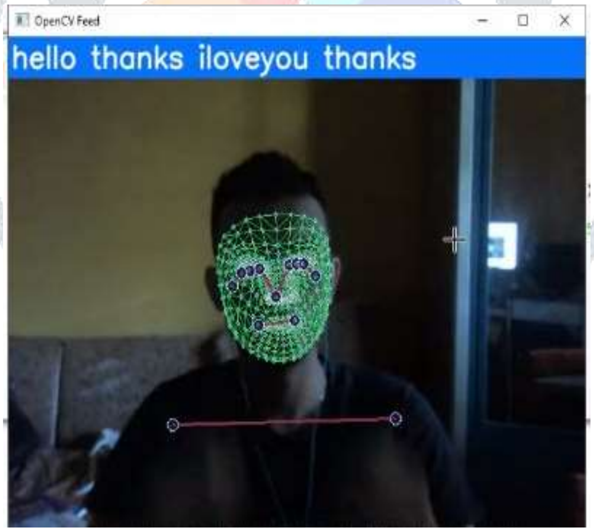


Fig. 5. Output of Module C

TABLE-I
EXISTING APPLICATIONS VS PROPOSED SYSTEM

Existing Applications	Proposed System
User interfaces are often limited and may require additional hardware	The proposed system is developed using OpenCV Library, LSTM model with Mediapipe kernel, and flutter plug-ins. It is designed to read and predict sign language gestures.
It requires a large amount of training data to achieve high accuracy.	It can be trained with a large amount of sign language data to improve accuracy
Some applications are limited to specific sign language	The proposed system can be use to support multiple sign language
The cost varies on the application.	Can be developed as an open source project, which makes it cost-effective.

V. CONCLUSION

This paper presents an automatic hand-sign language translator - a critical system for mute/deaf individuals. Expected requirements and level of performances of such a system are addressed here. The paper lists system components and extends the explanation of software. The software part of the system is extensively elaborated to include simple system initialization and recognition algorithms. The paper addresses the challenges of identifying ambiguous measurements and proposes respective technical solutions.

It is evident from experimental results that the system has the potential to help targeted individuals and communities. Especially that the system was able to recognise most of the letters. We look forward to adding the remaining dataset to the system and better the system performance.

REFERENCES

[1] Anand and N. P. Singh, "Sign Language Recognition System using TensorFlow Object Detection API," International Journal of Engineering and Advanced Technology (IJEAT), vol. 9, no. 6, pp. 2009-2016, 2020.

[2] Abhinav, V. Prasad, M. Sivakrishna, B. S. Sai, and P. Akshaya, "Sign Language Translator Using Machine Learning," International Journal of Advanced Science and Technology, vol. 29, no. 8, pp. 2089-2095, 2020.

[3] S. Naveen Kumar, S. S. Manjunatha, and B. Shivaprakash, "Real-Time Sign Language Detection using TensorFlow, OpenCV and Python," International Journal of Innovative Technology and Exploring Engineering (IJITEE), vol. 10, no. 9, pp. 59-64, 2021.

[4] P. M. Prabhu and M. Muthuselvam, "BIM Sign Language Translator Using Machine Learning (TensorFlow)," International Journal of Engineering Research and Technology (IJERT), vol. 10, no. 8, pp. 353-357, 2021.

[5] Jindal, A., Bansal, A., & Kumar, S. (2020). Sign Language Recognition using Convolutional Neural Networks. In 2020 International Conference on Computer Communication and Informatics (ICCCI) (pp. 1-6). IEEE.

[6] Ullah, M. A., Al-Maadeed, S., & Al-Emadi, N. (2018). Sign Language Recognition Using Convolutional and Recurrent Neural Networks. In 2018 International Conference on Electrical and Computing echnologies and Applications (ICECTA) (pp. 1-5). IEEE.

[7] Bao, X., Lin, J., Zhang, Y., & Cao, X. (2018). Real-time American Sign Language Recognition System using Deep Learning. In 2018 IEEE International Conference on Image Processing, Applications and Systems (IPAS) (pp. 11-16). IEEE.

[8] Li, H., Zhang, Y., Liu, X., & Liu, B. (2020). Real-time Chinese Sign Language Recognition System Using Deep Learning. IEEE Access, 8, 128862-128871.

[9] Huang, M., Wu, Y., Wang, Z., Zhang, Y., & Chen, W. (2021). English Text to American Sign Language Video Translation. In 2021 IEEE International Conference on Multimedia and Expo (ICME) (pp. 1-6). IEEE.

[10] Ananthi. S and Dhanalakhsmi P., “Survey about Speech Recognition and Its Usage for Impaired (Disabled) Persons)”, International Journal of Scientific & Engineering Research, Volume: 04, Issue: 02,February-2013 1 ISSN 2229-5518

[11] Wandera Roberts Otyola, Grace Milly Kibanja and Anthony M. Mugagga, “Challenges Faced by Speech Impaired Students at Makerere and Kyambogo Universities”, Makerere Journal of Higher Education, ISSN: 1816-6822; 9 (1) (2017) 75 – 86 [Online]