International Journal of Information Technology, Research and Applications (IJITRA)

Vishvanathan V, Vignesh J, Vinoth R, (2025). Streamlining Sign Language Translating Using OpenCV and Mediapipe through Mobile Application, 4(Special Issue), 42-50.

ISSN: 2583-5343

DOI:10.59461/ijitra.v4iSpecial Issue.124

The online version of this article can be found at:
https://www.ijitra.com/index.php/ijitra/issue/archive

Published by:
PRISMA Publications

International Journal of Information Technology, Research and Applications (IJITRA) is a journal that publishes articles which contribute new theoretical results in all the areas of Computer Science, Communication Network and Information Technology. Research paper and articles on Big Data, Machine Learning, IOT, Blockchain, Network Security, Optical Integrated Circuits, and Artificial Intelligence are in prime position.

https://www.prismapublications.com/

Streamlining Sign Language Translating Using OpenCV and Mediapipe through Mobile Application

Vishvanathan V[1], Vignesh J[2], Vinoth R[3]

[1-3]Department of Artificial Intelligence and Data Science, Rajalakshmi Institute of Technology, Chennai, India

Article Info

Article history:

Following professorial review at Rajalakshmi Institute of Technology, Chennai, India, the Department of Artificial Intelligence & Data Science has submitted the selected article for publication.

Received December 20, 2024
Accepted April 20, 2025

Keywords:

Sign Language Translation
Mobile application
OpenCV
Real-time Translation
Mediapipe

ABSTRACT

Sign language is the natural language used by Deaf and Mute people for communication, which is mainly expressed through hand gestures.This study is about creating an application for Deaf and Mute People which uses media Pipeline and OpenCV, The processing includes extract frame frame from the feed, process the image, detect hand landmarks, classification, conclude the output and The result is displayed as a text along with the overlay for the hands and the bounding box. This research studies whether it is possible to build such an application for the mobile platform with offline features.

Corresponding Author:

Vishvanathan V
Department of Artificial Intelligence and Data Science
Rajalakshmi institute of Technology
Chennai
India

Email: vishvanathan.v.r@gmail.com

1 Introduction

Sign language acts as a communicative tool for people who are unable to speak and people who are unable to hear, They are able to convey their thoughts, feelings, and ideas through gestures .the main objective of this study is to create an application for Deaf and Mute People which uses media Pipeline and OpenCV. The solution proposed here is an algorithm which can extract frames from the feed, process the image, detect hand landmarks, classification, conclude the output and The result is displayed as a text along with the overlay for the hands and the bounding box. In India over 1.5 million people use sign language for communication and In that population over 55% of the people can access smartphones, which can be other ways used as a communication tool with this algorithm. Initially to speak a new language 300 words will be adequate, so building an application requires most commonly used 300 words from a language, frequently used words from the user can be stored for personalization. Using API from other applications helps us to give input as text from gestures.

2 Method

2.1 Sign Language

Sign language is used by deaf and dumb people to communicate with others. There are different variants of sign languages in different countries such as American Sign Language, British Sign Language, Indian Sign Language, Chinese Sign Language.

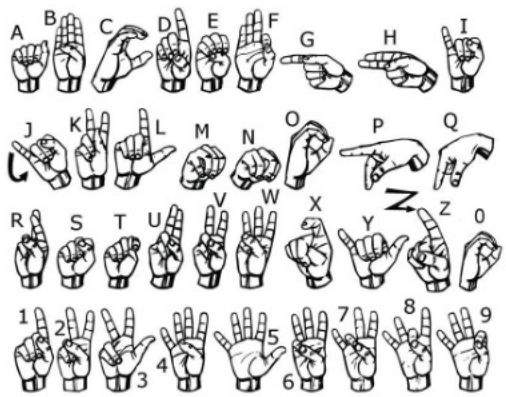

Figure 1: American Sign Language

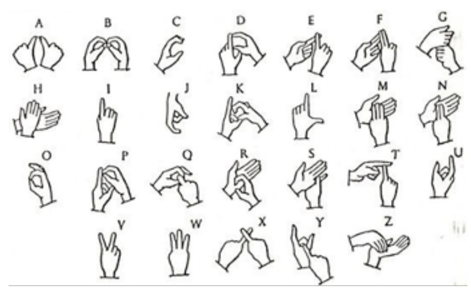

Figure 2: Indian Sign Language

Figure 3: Arabic Sign Language.[19]

As seen from figures 1, 2 and 3 we can notice that various sign languages have different signs for the same letter. Indian sign language is the method that is widely used in parts of India.[4]

2.2 Classification:

The input for the application can be derived from either the camera, a video or an image from the storage of the smartphone. Once the input is specified, the application begins a process of preprocessing the image or individual frames in the case of a video or a real time input through a camera. The preprocessed images are then handled by the Mediapipe's Hands module and OpenCV.[7] The connected landmarks are then joined and overlaid on top of the input stream.

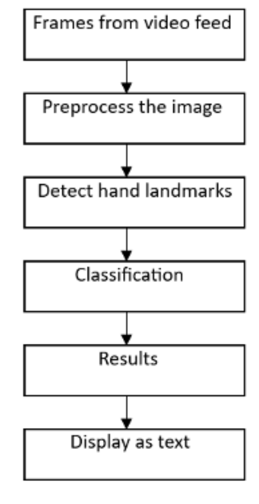

Figure 4: Classification process of image

We use OpenCV in conjunction with the Mediapipe's Hands model to classify the hands in the input stream and their positions. As OpenCV is lightweight in size , efficient and low resource requirement, We use it for this application.[2] The Mediapipe's Hands model is used for detecting the landmarks in the input stream's gestures which are used to detect the corresponding symbol.[5] OpenCV is used for preprocessing the image which will then be relayed to the Mediapipe model as Mediapipe required the images to be detected in a RBG format. We also resize the images to be in a uniform size. We will be resizing the images to be in 640x640 in size to be in equal and uniform dimensions for the model to analyze.[3]

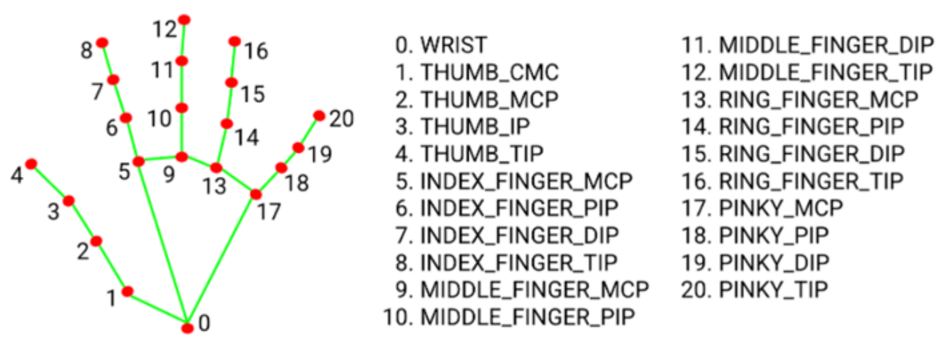

Figure 5: The typical landmarks used in Mediapipe

Once the image or a frame from an input stream is preprocessed, it will be fed into the Mediapipe model.[6] If any hands are detected in the image, Mediapipe's hands model extracts the landmarks from the image 6.[2] Each hand has a set of 21 landmarks as seen in figure 5.

Figure 6: The outputs of the Model

We use the 21 3D landmarks[6] from the model to detect the gesture in the input image. The handedness output is used for detecting whether it is the left hand or right hand in the image which will largely not be utilized except for the gestures which require both hands such as the gesture for opening a book in ASL.[4]

The landmarks extracted from the image are then analyzed by their relative positioning to each other. The landmarks are then mapped to a symbol in the database by the model based on their relative positioning. The results are then displayed on the application. We will be utilizing the Random Forest Classifier model for classifying the images. The random forest classifier has a 97% positive identification chance.

Few of the most utilized words and signs will be favoured for the model first to be able to be recognised first more than those not used as often. Once the classes which are marked as "Common use" are not the match of the input hand sign, Other classes in the database will be searched by the model. This is accomplished through the use of weights during the training phase. The resultant letter, word, or symbol is displayed in the application and the corresponding result is then added to the database. If this class is then is the output of several more searches, it is added to the "Common use" class list for more efficient searches. The classes are handled through a local storage of the OpenCV model required for classification which gets periodically updated based on user feedback and accuracy in detecting the various gestures used in the sign languages.

2.3 New Word Training

The user can upload a photo/video of a new sign word in the app. This photo/video is sent to the Firebase database, where it is stored along with the user's metadata.[1] The media file is then processed using OpenCV for the following tasks

1.Resizing
2.Background removal
3.Normalization
4.Augmentation

After processing, the photo/video is sent to MediaPipe to extract 21 key hand landmarks, capturing the position and orientation of the hand in 3D space. Once the data is in the correct format it is stored in the database, we begin training the model based on two types of signs:

1. Static Signs
2. Dynamic Signs

The model goes through continuous training to improve accuracy. Once trained the model is deployed to the app for real-time recognition of new sign words.

Flowchart:

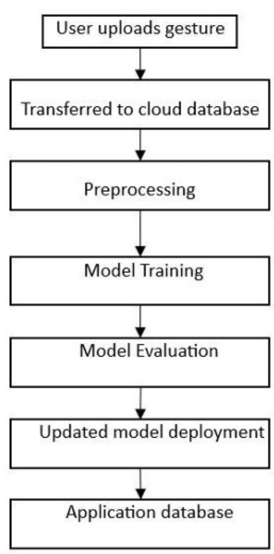

Figure 7: New Word Training

2.3.1 User Uploads Sign Image/Video

The user captures and uploads the image/video of a sign along with metadata through the app.

2.3.2 Store in Cloud Database

The image/video and associated metadata are stored in cloud database.For this we use Firebase database.[18]

1. Firebase storage for media
2. Firestore for metadata.[11, 12]

2.3.3 Preprocessing with OpenCV

The uploaded image/video is preprocessed using openCV. The preprocessing steps include
1.Resizing: Standardizing image size
2.Normalization: Scaling pixel values
3.Background Removal: Isolate the hand area

2.3.4 Mediapipe Hand Tracking

It is explained in the above section 2.2

2.3.5 Store Extracted image/video in Database

The extracted hand landmarks are stored in the database along with metadata for continuous training.

2.3.6 Model Training(CNN/RNN)

The stored data are used to train the sign language recognition based on two types.[8]
1. Static Signs : These are signs where the hand positions does not change over time (i.e) taking a photo of hand gesture.
2. Dynamic Signs : These are signs that involve movement or a sequence of actions (i.e) like a short video of showing hand movements.

2.3.7 Deploy Model to App

The updated model is deployed to the mobile app allowing the deaf and dumb peoples to real time sign recognition.[12, 13]

2.3.8 Real Time Sign Recognition

The application uses the trained model to recognize signs in real time from live camera input.[8, 9]

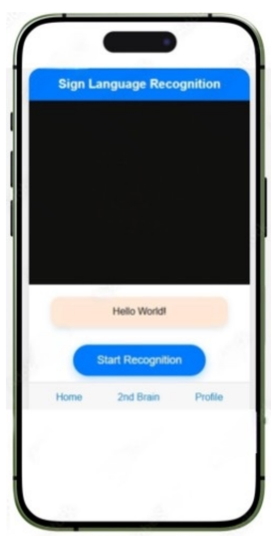

Figure 8: The home screen of the application

Figure 9: Translation screen

The figures 8 and 9 demonstrate the usage of the application from the user end.

3 Results and Discussion

The application proposed is technologically feasible. The selected tools include the OpenCV model which is trained through various datasets. The trained model has the following confidence graph.

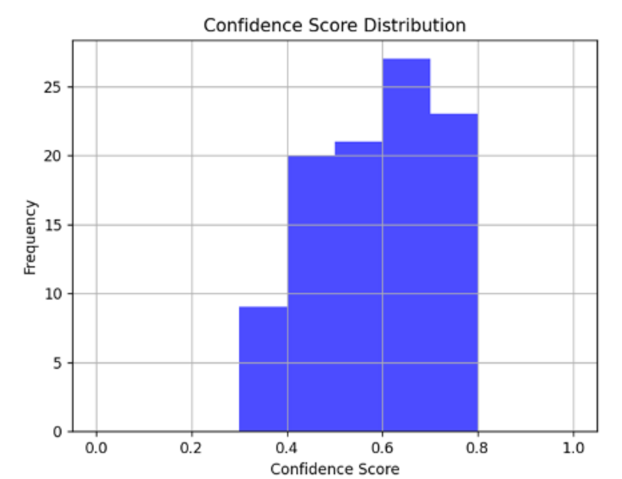

Figure 10: Confidence score of detection model

As seen from figure 10, The resulting model can recognize the gestures with a peak confidence of 0.8 with an average of 0.7 in most cases. The application can detect the correct gestures from a stream of input.

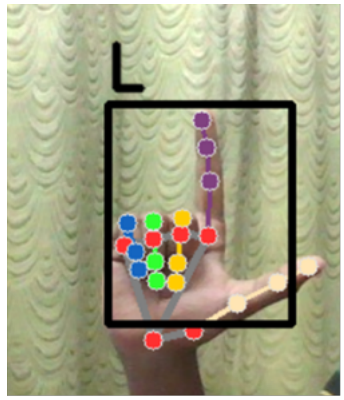

Figure 11: An example output of the model

The result is displayed as a text along with the overlay for the hands and the bounding box. The landmarks are clearly identified by different colors and then joined together. The application, compared to the currently available apps on various mobile platforms, offers an offline mode with multilingual support which is not provided by many of the applications in the store currently. Future scope of this study will be the inclusion of detection of facial features for easier detection of gestures.

3.1 Applications

3.1.1 Emergency Services

Enable easy communication between deaf/mute peoples and emergency responders in crucial time and accurate assistance during emergencies. By using these services translate sign language or text into speech and vice versa enable deaf/mute people to communicate their needs and understand instructions from responders.

3.1.2 Legal Services

In legal service, sign language animation technology can help deaf/mute people to understand legal documents, court proceedings and consultations by translating them into sign language.

3.1.3 Travel and Tourism

Enable effective communication between deaf/mute peoples and service providers is necessary for a smooth travel experience.By using this technology translate sign language or text into speech and vice versa, deaf/mute travelers can easily communicate with hotel staff, tour guides, airport personnel and other service providers.This technology allows them to make inquiries, request assistance and understood important information such as directions or announcements, enable a more accessible and enjoyable travel experience.

3.1.4 Classroom and college

By using this service, educational materials such as notes and textbooks can be converted into sign language animations, making it easier for deaf/mute students to understand the content.This technology helps deaf/mute students to understand the concept very well.

3.1.5 Workplace Meeting

By sign language animation technology translates spoken discussions into sign language, allowing deaf/mute employees to follow meetings and participate fully. This ensures equal communication during workplace discussions and presentations.

3.1.6 Public Announcements

In public spaces like airports and railway stations, sign language animation technology can convert important announcements into visual sign language and enabling mute people to receive correct information in real time.

4 Conclusion

Based on the result of the project, we can conclude that the OpenCV and Mediapipe Hands model with python can be used for detecting sign language gestures. This proposed application will utilize the firebase database to store and update its model for offline use. Using this application, vocally and hearing-impaired people will be able to communicate efficiently through the device which they already own. This improves their self-dependence and paves the way for increased accessibility to those with disabilities.

ACKNOWLEDGEMENTS

We express our utmost gratitude to Mrs. Merlin from the department of Artificial Intelligence and Machine Learning at Rajalakshmi Institute of Technology for their guidance and valuable insights.

References

[1] Sarthak, S., & Vinayak, R. (2024). Whisperings from the ordinary: A conversational artificial intelligence (AI) approach to bridging the human-object divide. International Journal of Information Technology, Research and Applications, 3(3), 1–6. https://doi.org/10.59461/ijitra.v3i1.89

[2] Anuar, A., Saipullah, K. M., Ismail, N. A., & Soo, Y. (2019). OpenCV based real-time video processing using Android smartphone. International Journal of Computer Technology and Electronics Engineering, 1(3). ISSN 2249-6343.

[3] Preet, A. (2022). Hand landmarks detection on an image using Mediapipe. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2022/03/hand-landmarks-detection-on-an-image-using-mediapipe/

[4] Handspeak. (n.d.). American Sign Language dictionary: Word 210. Handspeak. https://www.handspeak.com/word/210/

[5] Triyono, L., Pratisto, E. H., Bawono, S. A. T., Purnomo, F. A., Yudhanto, Y., & Raharjo, B. (2018). Sign language translator application using OpenCV. IOP Conference Series: Materials Science and Engineering, 333, 012109. https://doi.org/10.1088/1757-899X/333/1/012109

[6] Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., Chang, C.-L., & Grundmann, M. (2020). MediaPipe Hands: On-device real-time hand tracking [Preprint]. arXiv. https://arxiv.org/abs/2006.10214

[7] Tayade, A., & Gupta, K. (2021). Real-time vernacular sign language recognition using MediaPipe and machine learning. International Journal of Research Publication and Reviews, 2(5). ISSN 2582-7421.

[8] Kumar, R., Bajpai, A., & Sinha, A. (2023). MediaPipe and CNNs for real-time ASL gesture recognition [Preprint]. arXiv. https://arxiv.org/abs/2305.05296

[9] Haldera, A., & Tayade, A. (2021). Real-time vernacular sign language recognition using MediaPipe and machine learning. International Journal of Research Publication and Reviews, 2(5). ISSN 2582-7421.

[10] Nikam, A. S., & Ambekar, A. G. (2016). Sign language recognition using image-based hand gesture recognition techniques. In 2016 IEEE International Conference on Global Trends in Signal Processing, Information Computing and Communication (ICGTSPICC). https://doi.org/10.1109/GET.2016.7916786

[11] Khawas, C., & Shah, P. (2018). Application of Firebase in Android app development: A study. International Journal of Computer Applications, 179(46). https://doi.org/10.5120/ijca2018917200

[12] Sproull, T., & Shook, D. (2023). Machine learning on the move: Teaching ML Kit for Firebase in a mobile apps course. Proceedings of the 2020 ACM SIGCSE Technical Symposium on Computer Science Education. https://doi.org/10.1145/3328778.3367022

[13] Dai, X., Spasić, I., Chapman, S., & Meyer, B. (2020). The state of the art in implementing machine learning for mobile apps: A survey. 2020 IEEE SoutheastCon. https://doi.org/10.1109/SoutheastCon44009.2020.9249652

[14] Nigam, C., Narang, M., Chaurasia, N., & Nanda, A. (2020). M-Health Android application using Firebase, Google APIs and supervised machine learning. In Smart Innovations in Communication and Computational Sciences (pp. [page range if known]). Springer. https://doi.org/10.1007/978-3-030-32523-7

[15] Cooper, H., Holt, B., & Bowden, R. (2011). Sign language recognition. In M. A. Ferrandez, O. Garcia-Serrano, & A. Montoyo (Eds.), Handbook of Pattern Recognition and Computer Vision (pp. 539–562). Springer. https://doi.org/10.1007/978-0-85729-997-0_27

[16] Varshney, S. (2016). Deafness in India. Indian Journal of Otology, 22(2), 73–76. https://doi.org/10.4103/0971-7749.182281

[17] Nashat, D., Shoker, A., Al-Swat, F., & Al-Ebailan, R. (2014). An Android application to aid uneducated deaf-dumb people. International Journal of Computer Science and Mobile Applications, 3(12). ISSN: 2321-8363.

[18] Chougale, P., Yadav, V., & Gaikwad, A. (2021). Firebase - Overview and usage. International Research Journal of Modernization in Engineering Technology and Science, 3(12). e-ISSN: 2582-5208.

[19] Nawras, M. (n.d.). Momo Sign Language Learning. https://www.momosignlanguagelearning.com/arabic-sample.html

BIOGRAPHIES OF AUTHORS

Vishvanathan V
Vishvanathan V is pursuing a bachelor degree in B.tech Artificial Intelligence and Data Science from Rajalakshmi Institute of Technology, Chennai, Tamil Nadu, from 2023. He aims to become a scientist in ISRO and He is interested in electronic and electrical related topics like IOT, electronic systems, embedded systems. He has a strong foundation in electronics and a forward thinking mindset. He believes in changing the world by creating a sustainable environment and he believes technology should be used for a meaningful purpose. He can be contacted at email: VISHVANATHAN.V.R@gmail.com

Vinoth R
Vinoth R is pursuing a bachelor degree in B.tech Artificial Intelligence and Data Science from Rajalakshmi Institute of Technology, Chennai, Tamil Nadu, from 2023. He is deeply interested in app development, with a strong focus on integrating data science, machine learning models, and financial technologies to build intelligent, user-centric applications. In addition, He is passionate about app marketing and distribution strategies, and He aspires to create innovative and viral apps that can reach and impact a global audience. He is also highly intrigued by the field of finance and enjoys exploring how data-driven insights can be applied to develop smarter financial solutions within mobile applications. He can be contacted at email: vinoth.r.2023.aids@ritchennai.edu.in

Vignesh J
Vignesh J is pursuing a bachelor degree in B.tech Artificial Intelligence and Data Science from Rajalakshmi Institute of Technology, Chennai, Tamil Nadu, from 2023. He aims to be a data analyst after graduating. He is well versed in various data analysis tools such as Power Bi, python and MongoDB. He is particularly interested in automation of data science projects with the help of machine learning. He is interested in leveraging AI To eliminate human errors and solve problems in daily life. He believes that while AI is very useful, it can never replace humans. He can be contacted at email: VIGNESHJ722@gmail.com