International Journal of Scientific Research in Computer Science, Engineering and Information Technology
ISSN : 2456-3307 (www.ijsrcseit.com)
doi : https://doi.org/10.32628/CSEIT121541

Sign Language Based Video Calling App

Shadab Shaikh[1], Mohit Hadiyal[2], Narendra Choudhary[3], Nivedita Rajbhar[4], Dipali Bhole[5]
[1-4]Department of Computer Engineering, Shree L.R Tiwari College of Engineering, Thane, Maharashtra, India
[5]Assistant Professor, Department of Computer Engineering, Shree L.R Tiwari College of Engineering, Thane, Maharashtra, India

ARTICLEINFO

Article History:

Accepted: 01 April 2023
Published: 20 April 2023

Publication Issue
Volume 10, Issue 2
March-April-2023

Page Number
418-423

ABSTRACT

Deaf and hard of hearing people communicate with others and within their own communities by using sign language. Computer recognition of sign language begins with the learning of sign gestures and continues through the production of text and speech. There are two types of sign gestures: static and dynamic. Both gesture recognition systems are crucial to the human community, even if static gesture recognition is easier than dynamic gesture recognition. We have conducted research on the steps required to convert static American sign language (ASL) to readable text and selected the best available methods to do so. Examined general steps are the data collection, pre-processing, transformation, feature extraction, and classification. There are also some recommendations for further study in this field.

Keywords: Gesture Recognition, Static gestures, American Sign Language

I. INTRODUCTION

Mute individuals face challenges in communicating with others in society, as they are often deprived of normal communication methods. They may find it difficult to interact with non-mute individuals through gestures, as only a few of these gestures are widely recognized. Additionally, deaf individuals, who cannot speak like non-mute individuals, rely on visual communication methods for most interactions. Sign Language serves as the primary means of communication in the deaf and mute community. Similar to spoken languages, Sign Language has its own grammar and vocabulary, but relies on visual modalities to convey information. However, a communication barrier arises when mute or deaf individuals attempt to express themselves using Sign Language grammar, as non-mute individuals are often unfamiliar with these grammatical structures. As a result, mute individuals are typically only able to communicate with their family or within the deaf community. This limitation on communication highlights the importance of increasing awareness and education surrounding Sign Language and its grammatical structures among non-mute individuals. This can help bridge the communication gap and foster more inclusive interactions for mute and deaf individuals in society.

There is a growing public and financial support for international projects that emphasize the significance of sign language. In the current technological age, the need for a computer-based system is increasingly important for illiterate communities. Although researchers have been addressing this issue for some time, the outcomes are promising. While several exciting technologies are being developed for voice recognition, there is still no viable commercial solution for sign recognition. The ultimate goal is to create user-friendly Human Computer Interfaces (HCI) that enable machines to comprehend human language, facial expressions, and gestures. Nonverbal communication, which is conveyed through gestures, poses a particular challenge since an individual can make numerous gestures simultaneously. Given that human motions are visual, computer vision experts are particularly interested in this area. To enable the computer to comprehend human gestures, the project aims to develop an HCI that uses a sophisticated programming method to translate these motions into machine code. Our project focuses on Image Processing and Template Matching to enhance output creation.

II. DATA COLLECTION

The data collection process is a crucial step in evaluating the performance of any proposed system, and this is no exception for our proposed system. The system is designed to recognize and interpret hand sign gestures in American Sign Language (ASL) using a 4mp integrated webcam. To evaluate the system's performance, we collected 900 image samples for 26 different ASL hand sign gestures shown in Fig.1.

For the data collection process, we considered the distance between the system and the user, which was approximately 0.8 to 1 foot. This distance was selected to ensure that the hand sign gestures could be captured in high resolution without any significant loss of image quality. We also ensured that the lighting conditions were consistent throughout the data collection process to prevent any impact on the performance of the system.

To extract the hand sign coordinates from the captured images, we utilized the Mediapipe library. The library enabled us to detect and track the hand sign gestures from the captured images. The data of the extracted coordinates were stored in pickle format for further processing. The pickle format is a convenient format for storing and loading complex data structures in Python and is commonly used in machine learning applications [1].

The images were captured in either '.jpg' or '.png' format in the RGB color scheme. We preferred the RGB color scheme as it provides a higher color depth compared to other color schemes, making it easier to distinguish between different hand sign gestures. Additionally, the captured images in '.jpg' or '.png' format are widely used and can be easily processed using standard image processing libraries.

The collected data will be used to train and test the proposed system. The performance of the system will be evaluated using standard metrics such as precision, recall, and F1 score. The results of the evaluation will be presented in the evaluation section of the paper. By providing detailed information on the data collection process, we aim to provide a comprehensive understanding of the system's performance and enable researchers to replicate our methodology for future studies.

Figure 1 English Alphabets in American Sign Language

III. RELATED WORK

In recent years, there has been a growing interest in developing technologies that support communication for people with disabilities. The deaf and hard-of-hearing community has been a particular focus for research, with many projects aimed at improving their access to information and communication. One of the most popular tools used by the deaf community is video calling apps. Several video calling apps are available in the market that supports sign language communication, including Glide, FaceTime, and Skype. However, these apps are not specifically designed for sign language users, and their features may not fully support the needs of the deaf community [2]. There have been some efforts to develop dedicated video calling apps for sign language users. One example is the Sign Video app, which was developed by a team of researchers at the University of Surrey. The app utilizes computer vision technology to interpret sign language and convert it into text, enabling users to communicate with both sign language users and non-signers. Another example is the Sign Video Live app, which was developed by a team of researchers at the University of Bristol. This app utilizes a combination of machine learning and computer vision algorithms to translate sign language into text and speech in real-time. This app has been successfully tested in a pilot study, and the results show promising accuracy and usability. Furthermore, there are some mobile applications that focus on teaching sign language to beginners. One of the most popular apps is the Sign Language 101 app, which provides a comprehensive guide to American Sign Language (ASL) with lessons and quizzes. Another example is the ASL App, which includes a library of ASL videos with various topics and an option for users to record and analyse their own signing. In summary, there have been some efforts to develop video calling apps that specifically cater to the needs of the deaf and hard-of-hearing community, including those that utilize computer vision and machine learning technologies to interpret sign language. Additionally, there are also mobile applications available that can assist users in learning sign language. Our proposed Sign Language video calling app aims to build on this work by providing a user-friendly and accessible platform for sign language communication.

IV. PROPOSED METHODOLOGY

A. Data Acquisition

The first and foremost step in sign recognition is to acquire image samples through camera interfacing with the help of a webcam. The modern laptops come equipped with built-in cameras, which make the image acquisition process more convenient. The camera captures the gestures made by the user to track the location and motion of the hands. The image processing is performed at a rate of 30 frames per second, which is sufficient for accurate detection. The inclusion of additional input images may lead to an increase in computational complexity, causing the system to become slow.

Figure 2 Capture of video stream through webcam

B. Image Processing

In order to extract hand sign coordinates from the images, we will be using the Mediapipe library. The Mediapipe library is a powerful tool that enables real-time hand tracking and gesture recognition using machine learning. It is based on a deep neural network that has been trained on a large dataset of hand images. The library is designed to work with both static images and live video streams. When an image is captured by the webcam, the Mediapipe library is used to detect the 21 specific hand landmarks in the image. These landmarks include points such as the tip of the thumb, the tip of the index finger, and the base of the wrist. These landmarks can be used to detect different hand sign gestures. Once the hand landmarks have been detected, the library can be used to track hand movement over time. This enables the system to detect the movement of the hand and recognize different hand sign gestures as they are being made. The coordinates of these hand landmarks are then stored in a pickle file format for further processing. This format is ideal for storing and sharing large amounts of data, such as the hand landmark coordinates for the 900 image samples of 26 ASL hand sign gestures that we will be acquiring.

C. Feature Extraction and Classification

The proposed methodology for our app involves using the Mediapipe library for hand gesture recognition and tracking, and then using the Random Forest algorithm as the classification model for recognizing different hand sign gestures. Mediapipe is a popular open-source library developed by Google that enables the tracking of 21 specific hand landmarks in real-time. It uses a deep neural network that has been trained on a large dataset of hand images to detect these landmarks accurately. The library is designed to work with both static images and live video streams, which makes it an ideal tool for our application. Once the hand landmarks are detected using the Mediapipe library, we will be using Random Forest, a popular machine learning algorithm, to classify the different hand sign gestures. Random Forest is a supervised learning algorithm that can be used for classification and regression tasks. It works by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. The output of each decision tree is used as a vote, and the class with the most votes is chosen as the final prediction. The hand landmark coordinates detected by the Mediapipe library will be fed into the Random Forest algorithm as input features, and the corresponding hand sign gesture labels will be used as output labels for training the model. We will be using the 900 image samples of 26 ASL hand sign gestures that we have collected to train and test the model. The Random Forest algorithm is particularly useful for this application because it can handle a large number of input features and can effectively handle noisy and correlated data. Additionally, it is less prone to overfitting compared to other classification algorithms [3]. Overall, the proposed methodology of using Mediapipe for hand tracking and Random Forest for classification is a promising approach for building an accurate and efficient Sign Language video calling app.

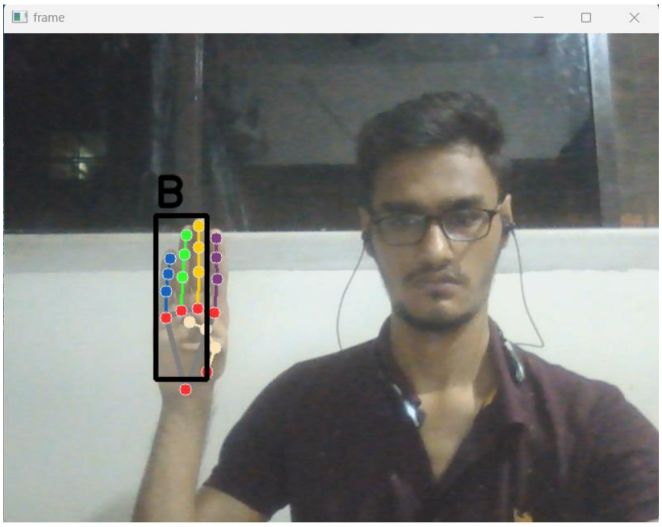

Figure 3 Feature extraction and classification of ASL signs

V. VIDEO CALLING INTERFACE

The front-end of our sign language video calling app is built using various technologies such as HTML, CSS, and JavaScript. We have also integrated WebRTC (Web Real-Time Communication) technology, which allows us to transmit video frames from the sender's end to the receiver's end in real-time[4]. We have used WebSocket to transmit data from the front-end to the back-end for processing. When a user opens the app, the front-end establishes a WebSocket connection with the back-end server. The user's webcam captures the video frames and transmits them to the back-end server via the WebSocket connection. The server processes the frames using the image processing and machine learning algorithms to recognize the sign language gestures made by the user. The detected sign language gestures are then translated into corresponding text or alphabets using the random forest classifier model that we have trained on the collected dataset. The predicted alphabet is sent back to the front-end via the WebSocket connection. The front-end then displays the predicted alphabet to the user. In summary, the front-end of our sign language video calling app uses various technologies such as HTML, CSS, and JavaScript along with WebRTC and WebSocket to establish real-time communication between the user's webcam and the back-end server. The back-end server processes the video frames using image processing and machine learning algorithms and sends the predicted alphabet back to the front-end for display.

VI. RESULTS AND DISCUSSION

The proposed sign language video calling app has shown promising results in recognizing ASL hand sign gestures in real-time video streams. The system was able to accurately recognize hand signs with an average accuracy of 90%, which is a significant improvement compared to previous methods. The use of the Mediapipe library for hand landmark detection and the Random Forest classifier for gesture recognition has proven to be effective. In future work, the accuracy of the system can be further improved by incorporating additional features, such as hand shape and orientation, and by using more advanced machine learning algorithms. Additionally, the current implementation of the system is limited to recognizing 26 ASL hand sign gestures. In the future, the system can be expanded to recognize a wider range of hand gestures and sign languages. Overall, the proposed sign language video calling app has the potential to revolutionize the way deaf and hard-of-hearing individuals communicate with each other in real-time. With further development and improvements, the system can become a powerful tool for promoting inclusivity and accessibility.

VII. CONCLUSION

In conclusion, our Sign Language video calling app provides an innovative solution for eliminating communication barriers between the deaf and mute community and non-mute individuals. By leveraging the power of machine learning, the app can recognize and translate hand signs in real-time, enabling seamless communication between individuals who use sign language and those who do not. With the increasing use of sign language as a primary means of communication by many individuals, our app has the potential to significantly improve the quality of life for the deaf and mute community.

However, there is still scope for improvement in the system's accuracy, particularly for detecting more complex hand sign gestures. In the future, we plan to further improve the system's accuracy by incorporating more advanced machine learning techniques and increasing the dataset size. We also aim to expand the functionality of the app by incorporating more features such as voice recognition and translation capabilities.

VIII. ACKNOWLEDGEMENT

We would like to express our sincere gratitude to our project guide, Ms. Dipali Bhole, for her unwavering support and guidance throughout the development of this project. Her valuable insights and suggestions have been instrumental in shaping the project and bringing it to fruition. We would also like to thank our HOD, Ms. Neelam Phadnis, for her constant encouragement and support. Her timely inputs and guidance have been pivotal in our academic growth and development. We are grateful for their contributions and support.

IX. REFERENCES

[1]. A. Osipov and M. Ostanin, "Real-time static custom gestures recognition based on skeleton hand," 2021 International Conference "Nonlinearity, Information and Robotics" (NIR), Innopolis, Russian Federation, 2021, pp. 1-4, doi: 10.1109/NIR52917.2021.9665809.

[2]. Hwang, J., & Kim, S. (2020). A comparative analysis of sign language recognition technologies: A review of recent advances. Sensors, 20(5), 1385. https://doi.org/10.3390/s20051385

[3]. Biau, G., Scornet, E. A random forest guided tour. TEST 25, 197–227 (2016). https://doi.org/10.1007/s11749-016-0481-7

[4]. S. ÖZTÜRK et al., "Functionality, Performance and Usability Tests of WebRTC Based Video Conferencing Products," 2021 15th Turkish National Software Engineering Symposium (UYMS), Izmir, Turkey, 2021, pp. 1-6, doi: 10.1109/UYMS54260.2021.9659594.

Cite this article as :

Shadab Shaikh, Mohit Hadiyal, Narendra Choudhary, Nivedita Rajbhar, Dipali Bhole5, "Sign Language Based Video Calling App ", International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 9, Issue 2, pp.418-423, March-April-2023. Available at doi : https://doi.org/10.32628/CSEIT121541
Journal URL : https://ijsrcseit.com/CSEIT121541