ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

PANTOMIME RECOGNITION USING MACHINE LEARNING

[1]S Ritesh Jaiswal, [2] Cherla Mounika, [3] Boddupalli Likitha, [4] Bojja Gopichand [1,2,3,4] B.Tech, Department of Computer Science and Engineering, TKR College of Engineering and Technology, Hyderabad, India.
Dr. Chaganti B N Lakshmi, Professor, Department of Computer Science and Engineering, TKR College of Engineering and Technology, Hyderabad, India

Abstract
People who are deaf or hard of hearing use sign language to interact with others and within their own groups. Learning sign motions is the first step in computer recognition of sign language, which continues when text and voice are produced. Static and dynamic sign motions are the two categories. Even though static gesture recognition is simpler than dynamic gesture recognition, both gesture recognition systems are essential to human society. Understanding and learning sign language takes a lot of practice, and not everyone will comprehend what the motions in sign language indicate. Convolutional neural networks are used in pantomime recognition to help humans learn sign language and convert it into letters and words.

Keywords: Machine learning, Pantomime, Image classification, Convolutional Neural Network

I. INTRODUCTION

Motivation

In our modern culture, it is important to interact with everyone, whether it be for fun or business. Every human being needs to communicate. However, individuals with speech or hearing impairments require a different form of communication than vocalization. They use sign language as a means of communication.

Problem definition

Sign language is also known as pantomime, which is the art or ability of utilizing gestures instead of speaking to convey emotions, actions, feelings, etc. However, sign language comprehension requires a lot of practice, and not everyone will be able to grasp the meaning behind the movements. There are also no dependable, portable tools available, thus learning sign language takes time. To successfully express their thoughts to others, hearing- or speech-impaired people who are fluent signers require a translator who is also fluent in sign language.

Limitations of existing system

The existing system helps hearing or speech disabled people to learn as well as translate their sign language to alphabets (A-Z) excluding J and Z. The system incorrectly predicts multiple signs, and some are labeled incorrectly. Due to the significant similarities between some letters and some digits, the recognition system with digits included had a reduced accuracy rate.

Proposed system

In the proposed system, we built a deep convolutional neural network (CNN) with additional layer batch normalization. The proposed system aids individuals who have difficulty hearing or speaking in learning new sign language and converting it into words and phrases. The model is able to transform the sign language images into alphabets (A-Z), numbers (1-10) and some words and sentences, including "thank you," "sorry," "ok," "best of luck," "is everything alright," "i love you," "rock and roll," and "space." This approach successfully classifies all signs with great accuracy. The system is tested using hyper parameters, which produce good accuracy.

II. LITERATURE REVIEW

People who have trouble hearing or speaking rely substantially on sign language in their everyday lives, according to work [1]. They may converse by using hand gestures. American Sign Language (ASL) has a high degree of complexity and is becoming more and more similar among classes, making it challenging to recognise. In [1], created a deep convolutional neural network to tackle the

problems of ASL alphabet recognition. A deep convolutional neural network-based method for ASL recognition is provided in this study. Because the effectiveness of the Deep CNN model depends on the volume of input data, the size of the training data set is artificially increased using the data augmentation approach. The results of the studies demonstrate that the recommended DeepCNN model produces accurate outcomes for the ASL dataset. Experiments demonstrate that the DeepCNN offers accuracy increases of 19.84%, 8.37%, 16.31%, 17.17%, 5.86%, and 3.26% when compared to a number of state-of-the-art techniques.

Sign language, which was developed with this objective in mind, allows communities of the deaf and dumb to communicate with one another and with the rest of society. The Leap Motion Controller (LMC) was used in [2] to construct a sign language recognition prototype due to society's regrettable lack of interest in learning and utilizing sign language. This study aimed to fully recognise American Sign Language (ASL), which consists of 26 letters and 10 numbers, in contrast to various other studies that merely provided ideas for ways to partially recognise sign language. While certain ASL letters are dynamic and need specific movements, the bulk of ASL letters are static and require no movement. This study further aimed to identify features from hand and finger motions in order to differentiate between static and dynamic gestures. Using a support vector machine (SVM) and a deep neural network (DNN), the experimental results show that the 26 letters have sign language recognition rates of 80.30% and 93.81%, respectively. However, recognition rates for a combination of 26 letters and 10 numbers are slightly lower (72.79% for the SVM and 88.79% for the DNN). Therefore, sign language recognition technology has great potential for bridging the social divide between the deaf and dumb cultures. In public locations like banks and post offices, the proposed prototype might possibly serve as a sign language interpreter for the blind and deaf.

In [3], researchers aim to precisely pinpoint unsegmented signals related to continuous sign language recognition (CSLR) from video streams. Despite the advances in deep learning techniques that have been made in this area, the bulk of them generally focus on using just one RGB component, such the full-frame image or the specifics of the hands and face. The paucity of data for the CSLR training technique places severe restrictions on the capacity to learn a variety of characteristics utilizing the video input frames. Additionally, using every frame in a video for the CSLR task may not yield the best results since each frame has a different amount of information, including fundamental properties in the inference of noise. They therefore offer a unique spatio-temporal continuous sign language recognition technique utilizing the attentive multi-feature network in order to enhance CSLR by including additional keypoint characteristics. They also emphasize several important elements simultaneously by using the attention layer in the spatial and temporal modules. Experimental findings from both the CSL and PHOENIX datasets reveal that the proposed technique beats state-of-the-art methods by 0.76 and 20.56 for the WER score, respectively.

Sign language is an effective means of human communication, and computer vision systems are the subject of extensive research. In the early studies on Indian Sign Language (ISL) recognition, a restricted subset of ISL signals were usually chosen for recognition, taking into account the recognition of substantially differentiable hand signs. In [4], stable static signs are modeled using convolutional neural networks (CNN) based on deep learning to understand sign language. For this study, 100 static signs comprising 35,000 sign photographs were collected from diverse users. The performance of the proposed system is evaluated on about 50 CNN models. The findings were also evaluated based on a number of optimizers, and it was discovered that the suggested technique had the maximum training accuracy, scoring 99.72% for coloured photos and 99.90% for grayscale images. The performance of the recommended system has also been evaluated using precision, recall, and F-score. The method also appears to be more successful than earlier efforts that merely took a few hand movements into consideration while distinguishing individuals.

Hand signals are a useful tool for human-to-human communication and have a wide range of applications. Since they are a natural way to interact, persons who have trouble speaking use them regularly to communicate. In reality, this group makes up just around 1% of Indians. This is the key argument in favor of why it would be extremely beneficial for these folks to incorporate a system that could translate Indian Sign Language. In [5], a technique using the Bag of Visual Words model

(BOVW), detects Indian sign language alphabets (A-Z) and numbers (0-9) in a live video stream and outputs the anticipated labels as both text and voice. Segmentation is carried out based on background removal and skin tone. Following the creation of histograms to map the indications with the proper labels, SURF (Speeded Up Robust Features) features are extracted from the photos. Convolutional neural networks (CNN) and support vector machines (SVM) are used for classification. An interactive graphic user interface (GUI) that is user-friendly is also created.

In [6], a dual leap motion controller-based system for Arabic sign language recognition is provided. They specifically put out the notion of using both front and side LMCs to overcome the problems of finger occlusions and lost data. For feature extraction, they selected a good set of geometric characteristics from both controllers, and for classification, they employ both a Bayesian technique with a Gaussian mixture model (GMM) and a conventional linear discriminant analysis (LDA) strategy. They provided a method for combining the data from the two LMCs that is evidence-based, especially the Dempster-Shafer (DS) theory of evidence. Two native adult signers provided data for 100 isolated dynamic Arabic signals. There were 10 observations altogether for each sign. The recommended architecture uses an ingenious technique to handle the case where one or both controllers have incomplete data. The recognition accuracy was around 92%. The proposed technique outperforms single-sensor approaches and contemporary glove-based systems.

III. DESIGN

The designing consists of mainly 3 modules named as Image capture and Pre-process module, Descriptor module and Classifier module.

Image capture and Pre-process module :

Computer vision systems must have an image capture and pre-processing module. This module is in charge of gathering raw picture data from a camera or other imaging equipment, processing it to weed out noise, account for lighting, and get it ready for further examination by more advanced computer vision algorithms. In order to capture photos in a certain format and resolution, the image capture component often includes interacting with the hardware, such as a digital camera. Image scaling, Colour space conversion, Noise reduction, Contrast improvement, and Image segmentation are the processes that commonly make up the pre-processing stage.

Descriptor module :

The descriptor module, which is utilized for feature extraction and matching in many computer vision systems, is a crucial element. We loaded the dataset and built a deep convolutional neural network (CNN) with additional layer batch normalization in this module. The loaded dataset is used to train the model. Testing the unobserved data provides a measurement of model accuracy.

Classifier module :

A machine learning model's classifier module is in charge of determining the class or category of a given input data point. In supervised learning, a classifier module is trained using labeled data, where the input data is linked to a predetermined class label. In the course of training, the classifier gains the ability to spot patterns or features in the input data that are characteristic of a specific class.

Data Flow Diagrams

A data flow diagram (DFD) is a straightforward graphical representation that may be used to illustrate a system's incoming data, the processing operations performed on those data, and the output data the system produces. It depicts the information flow for any system or process, showing the inputs and outputs of the data processing.

DFD level-0

Another name for Level 0 is a context diagram. It provides a general summary of the entire system or process that is being studied or modeled. It depicts the system as one cohesive, high-level process that interacts with outside entities. The DFD level-0 for the proposed system is depicted in Fig 1.

DFD level-1

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

The Context Level Diagram is shown in more detail in Level 1. The Context Diagram's high-level process is divided into its component parts. It draws attention to the primary duties performed by the system. The DFD level-1 for the proposed system is depicted in Fig 2.

DFD level-2

Level 2 offers a more thorough look than Level 1. To provide the essential degree of detail regarding how the system works, extra language may be needed. The DFD level-2 for the proposed system is depicted in Fig 3.

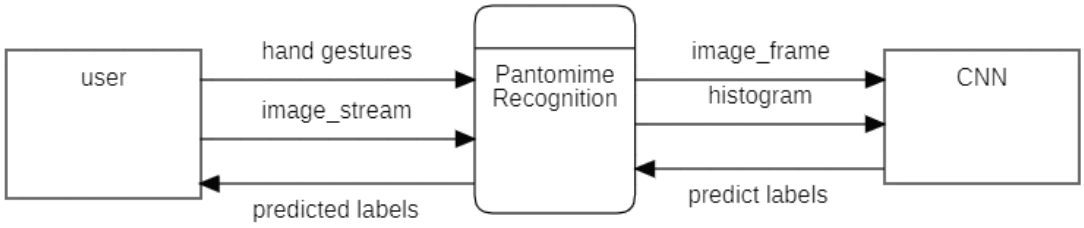

Fig. 1 DFD level-0

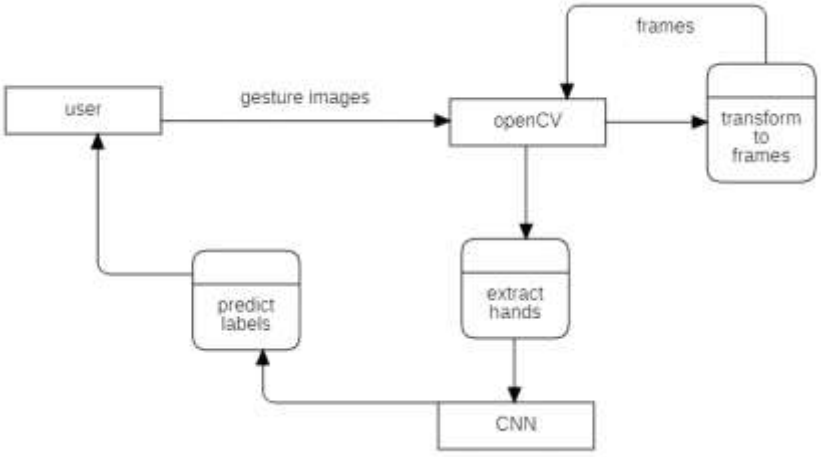

Fig. 2 DFD level-1

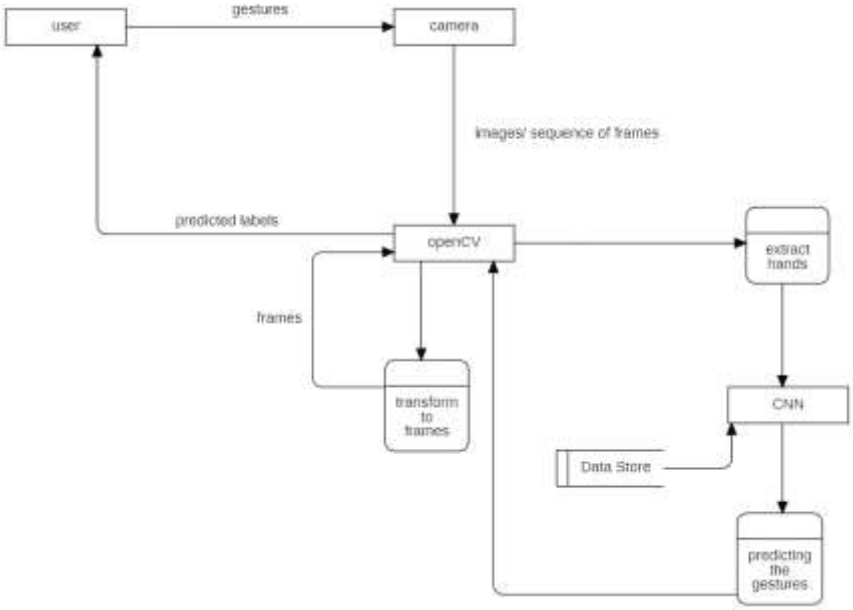

Fig. 3 DFD level-2

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

Use Case Diagram

A use case diagram displays a group of actors (a particular type of class) and their relationships. Use case diagrams deal with a system's static use case perspective. The use case diagram for the proposed system is shown in the figure 4.

➢ Use case subject is Pantomime recognition.

➢ Actors used are user and system.

➢ There are 8 use cases/flow of events namely Start Webcam, Capture Image, Capture Gesture, Translate Gesture, Extract Feature, Match Features, Recognizing Gestures and Display Result.

➢ The user can access the use cases start webcam and display result whereas the system can access all the use cases.

➢ Control enters the model when we run the code and exits when the label is displayed.

Table 1 Use case scenario for Pantomime recognition system

Use case subject	Pantomime recognition
Participating Actors	User, System
Flow of Events/ Use cases	Start Webcam Capture Image Capture Gesture Translate Gesture Extract Feature Match Features Recognizing Gestures Display Result
Entry Condition	Run The Code
Exit Condition	Displaying The Label

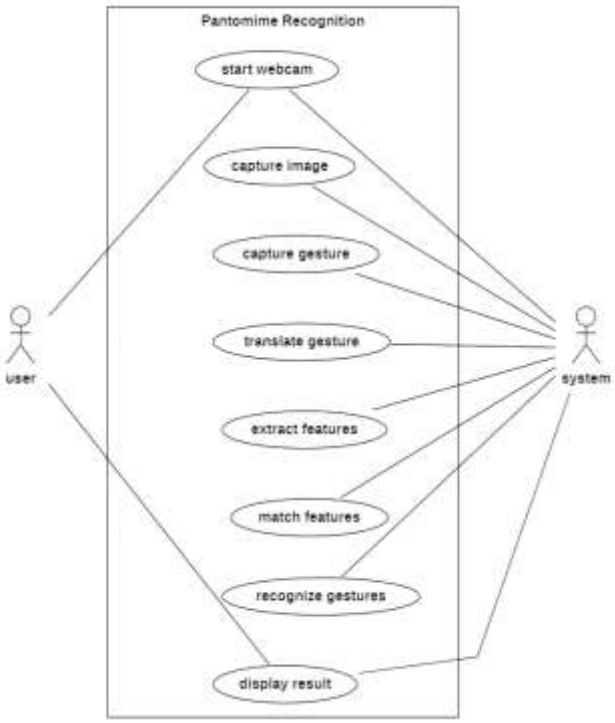

Fig. 4 Use case diagram

Algorithm

Step-1 : Capture the images and preprocess those images.

Step-2 : Load the data from the dataset.

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

Step-3 : Display the images.
Step-4 : Create a CNN model.
Step-5 : Train the model.
Step-6 : Predict the labels for test data.
Step-7 : Compute accuracy of the model.

Sample Data

➢ The Indian Sign Language(ISL) dataset contains test set and training set images of the signs for Indian sign language which includes 1 to 10 numbers, A - Z alphabets and some words/sentences.

➢ For each category or class,

○ Test set consists of 200 images.

○ Training set consists of around 1500 images.

➢ The data available in this dataset is already preprocessed and there is no need for further preprocessing on it.

➢ The words/sentences in our dataset are:

space	ok
best of luck	rock and roll
thank you	i love you
sorry	is everything alright

IV. IMPLEMENTATION and RESULTS

Method of Implementation

Tensorflow

A widely recognised open-source software library for applications involving machine learning and artificial intelligence is TensorFlow. For creating and training various neural network architectures, such as convolutional neural networks, recurrent neural networks, and generative adversarial networks. It offers a versatile and effective platform. The calculations of a machine learning model are represented by TensorFlow using a data flow graph. Each node in this graph corresponds to a mathematical operation, and each edge denotes the movement of data between operations. Many different companies and research areas utilize TensorFlow to do tasks including speech and image recognition, natural language processing, robotics, and recommendation systems.

Keras

The Python-based high-level neural network API Keras may be used with TensorFlow and other common deep learning frameworks. It was made to make the process of building deep learning models simple and require little code. Developers may concentrate on the model architecture and experiment with alternative configurations by using Keras' user-friendly and modular interface for developing deep learning models instead of worrying about the implementation's minute details. Convolutional, recurrent, and pooling layers, among others, are just a few of the pre-built layers offered by Keras that may be quickly added to the model. Additionally, it provides a number of optimisation methods, loss functions, and metrics, which makes it simple to adapt the training procedure for a particular purpose.

OpenCV

OpenCV (Open Source Computer Vision) is an open-source library of computer vision and machine learning algorithms that can be used to develop real-time computer vision applications. It was originally developed by Intel in 1999 and later maintained by the OpenCV Foundation.

OpenCV supports a wide range of programming languages such as Python, C++, Java, and MATLAB. It provides a set of functions and tools for image processing, feature detection, object recognition, face detection, and tracking, among other things. Some of the most common features of **OpenCV include:**

➢ Image and video I/O

➢ Image filtering and manipulation

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

- Object detection and recognition
- Camera calibration and 3D reconstruction
- Feature detection and tracking
- Machine learning and deep learning support
- GUI development

We interpret hand motions using the CNN method. The CNN technique is used to categorize a picture based on multiple traits and make it feasible to distinguish it from the classes to which it belongs. The CNN technique operates by transferring the input images between various layers.

Output Screens

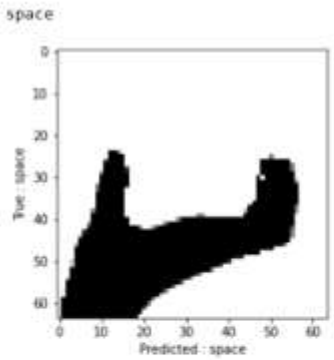

Fig. 5 space

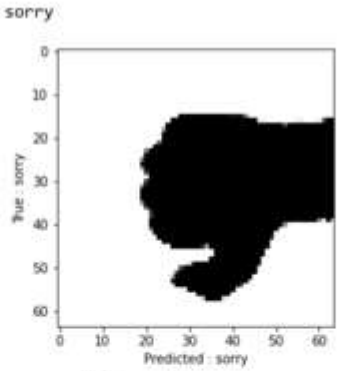

Fig. 6 sorry

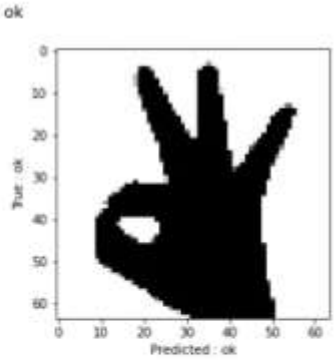

Fig. 7 ok

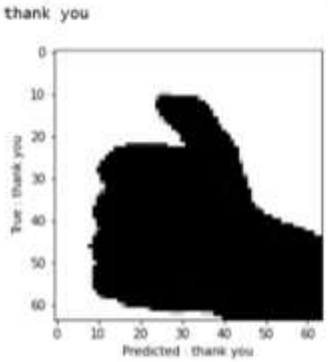

Fig. 8 thank you

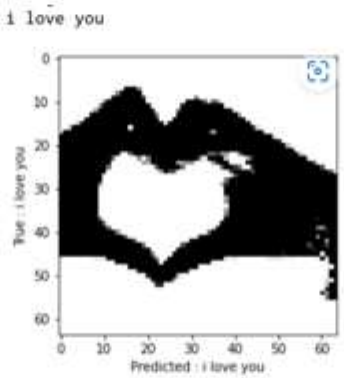

Fig. 9 i love you

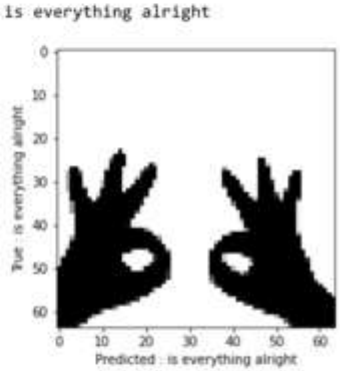

Fig. 10 is everything alright

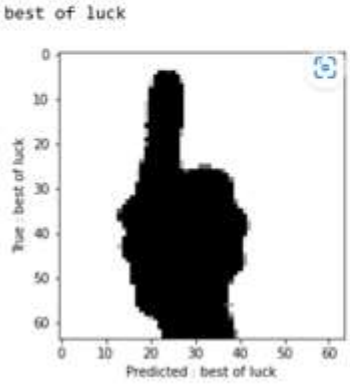

Fig. 11 best of luck

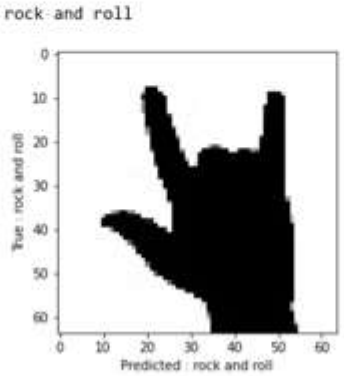

Fig. 12 rock and roll

Result Analysis :

We may deduce from the output screens shown above that our CNN model predicts the visual signs accurately. The output screens' graph displays the image's actual label as well as its predicted label.

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

All 44 signs are predicted by the CNN model, although not all of their images. It can accurately anticipate between 80% and 90% of the images across all signs.

V. TESTING AND VALIDATION

Test Case Scenarios

Test Case 1

During this, 12 epochs done, and the results are shown in figure 13. Observed low model accuracy and considerable model loss.

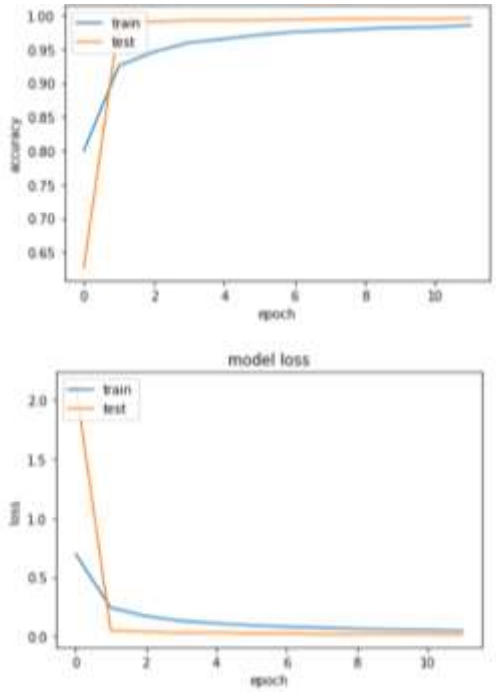

Fig. 13 Accuracy of Test case 1

Test Case 2

During this, 15 epochs done, and the results are shown in figure 14. With this Observed high model accuracy and minimal model loss.

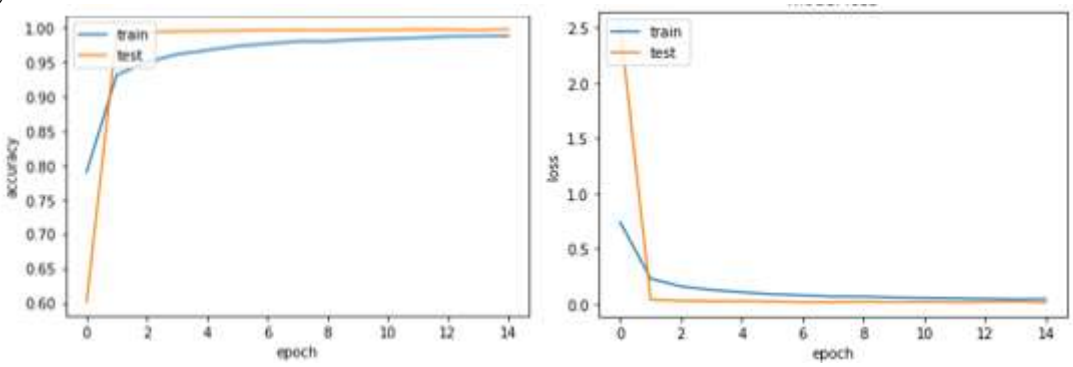

Fig. 14 Accuracy of Test case 2

Test Case 3

During this, 20 epochs done, and the results are shown in figure 15. Compared to the other test cases, obtained very high model accuracy and little model loss.

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

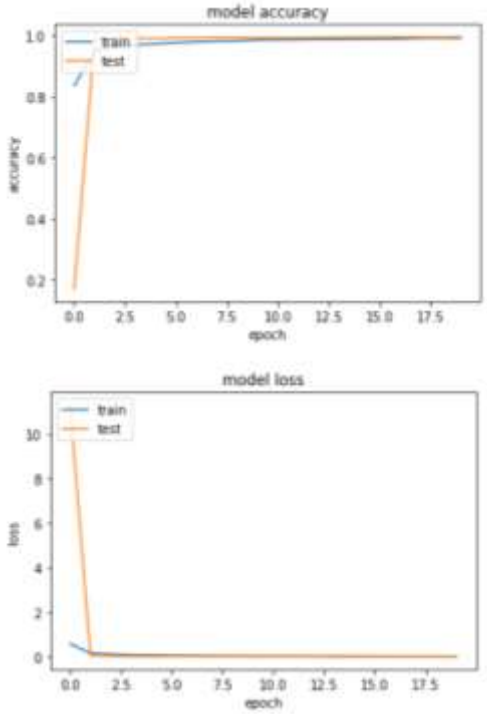

Fig. 15 Accuracy of Test case 3

Validation

Table 2 Validation

Epoch	Accuracy	Loss	Val_accuracy	Val_loss
1	0.8371	0.5939	0.1709	11.3018
8	0.9833	0.0541	0.9951	0.0234
12	0.9876	0.0408	0.9946	0.0408
16	0.9900	0.0334	0.9962	0.0185
18	0.9924	0.0262	0.9962	0.0174
20	0.9924	0.0264	0.9966	0.0194

From the table above, it is concluded that as the number of epochs grows, the model's accuracy increases. The accuracy ranges from 0.8371 in the first epoch to 0.9900 in the sixteenth. The accuracy after 20 epochs is 0.9924.

VI. CONCLUSION

The purpose of this paper is to identify the hand gestures and sign language used by various deaf and dumb individuals. In this paper, a useful technique for identifying ISL letters, numbers and words used in daily life is described. Regarding changes in parameters like the number of layers and epochs, the system produces the greatest training and validation accuracy of 98% and 99%, respectively. The proposed technique will have reater significance in the field of sign language recognition. Future work will focus on creating a sign language recognition system for hearing-impaired individuals as well as a real-time, highly accurate, and reasonably priced system for other fields that recognise dynamic gestures or movies. It can be extended to word and sentence based recognition.

ISSN: 2278-4632
Vol-13, Issue-07, No.04, July : 2023

VII. REFERENCES

[1] Abdul Mannan, Ahmed Abbasi, Abdul Rehman Javed, Anam Ahsan, Thippa Reddy Gadekallu and Qin Xin, "Hypertuned Deep Convolutional Neural Network for Sign Language Recognition", Computational Intelligence and Neuroscience, Volume 2022, Article ID 1450822, 10 pages.

[2] Teak-Wei Chong and Boon-Giin Lee, "American Sign Language Recognition Using Leap Motion Controller with Machine Learning Approach", Sensors 2018, 18 pages.

[3] Wisnu Aditya, Timothy K. Shih, Tipajin Thaipisutikul, Arda Satata Fitriajie, Munkhjargal Gochoo, Fitri Utaminingrum, Chih-Yang Lin, "Novel Spatio-Temporal Continuous Sign Language Recognition Using an Attentive Multi-Feature Network", Sensors 2022, 22 pages.

[4] Ankita Wadhawan, Parteek Kumar, "Deep learning-based sign language recognition system for static signs", Neural Computing and Applications (2020) 32:7957–7968.

[5] Shagun Katoch, Varsha Singh, Uma Shanker Tiwary, "Indian Sign Language recognition system using SURF with SVM and CNN", Array 14 (2022), 100141, 9 pages.

[6] Mohamed Deriche , Salihu O. Aliyu , and Mohamed Mohandes, "An Intelligent Arabic Sign Language Recognition System Using a Pair of LMCs With GMM Based Classification", IEEE Sensors Journal, Vol. 19, No. 18, September 15, 2019, 12 pages.

[7] Aurelijus Vaitkevičius, Mantas Taroza, Tomas Blažauskas, Robertas Damaševičius, Rytis Maskeliunas and Marcin Wozniak, "Recognition of American Sign Language Gestures in a Virtual Reality Using Leap Motion", Applied Sciences 2019, Article ID 9030445.

[8] Jungpil Shin, Akitaka Matsuoka, Md. Al Mehedi Hasan and Azmain Yakin Srizon, "American Sign Language Alphabet Recognition by Extracting Feature from Hand Pose Estimation", Sensors 2021, 21 pages.

[9] Ying Ma, Tianpei Xu, Kangchul Kim, "Two-Stream Mixed Convolutional Neural Network for American Sign Language Recognition", Sensors 2022, 22 pages.

[10] Hema B N, Sania Anjum, Umme Hani, Vanaja P, Akshatha M, "Survey on Sign Language and Gesture Recognition System", International Research Journal of Engineering and Technology (IRJET), Volume: 06, Issue: 03, Mar 2019.

[11] HONGGANG WANG , MING C. LEU AND CEMIL OZ, "American Sign Language Recognition Using Multi-dimensional Hidden Markov Models", Journal of Information Science and Engineering, 22, 1109-1123 (2006).

[12] Atreya Bain, Shiwam Birajdar, Prof. Manonmani S, "Sign language Recognition System using Machine Learning", International Research Journal of Engineering and Technology (IRJET), Volume: 08, Issue: 08, Aug 2021.

[13] Roshankumar L, Suriya S, Madesh G, Dr. W Gracy Theresa, "Sign Language Prediction using CNN", International Research Journal of Modernization in Engineering Technology and Science, Volume:04/ Issue:05/ May-2022.

[14] Kushal R Kabbathi, Bacha Ruthvik, Ayush Allen Louis, Sidhant Pareek, Anusha LS, "Detection of Sign Language using Machine Learning", International Research Journal of Modernization in Engineering Technology and Science Volume: 04/ Issue: 07/ July-2022.

[15] Ms. Ruchika Gaidhani, Ms. Payal Pagariya, Ms. Aashlesha Patil, Ms. Tejaswini Phad, Mr. Dhiraj Birari, "Sign Language Recognition using Machine Learning", International Research Journal of Engineering and Technology (IRJET), Volume: 09/ Issue: 01/ Jan 2022.