A Project Report

on

SIGN LANGUAGE GESTURE RECOGNITION FROM VIDEO SEQUENCE USING TENSORFLOW.JS

Submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Technology in Computer Science and Engineering

GALGOTIAS UNIVERSITY

(Established under Galgotias University Uttar Pradesh Act No. 14 of 2011)

Under The Supervision of
Dr. Pooja Singh
Associate Professor

Submitted By

Sarthak Luthra
19SCSE1180075

Shahreen Ali
20SCSE1180053

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING / DEPARTMENT OF COMPUTERAPPLICATION
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
March, 2023

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

We hereby certify that the work which is being presented in the project, entitled **"SIGN LANGUAGE GESTURE RECOGNITION FROM VIDEO SEQUENCE USING TENSORFLOW.JS"** in partial fulfillment of the requirements for the award of the Bachelor of Technology submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of January, 2023 to May, 2023, under the supervision of Dr. Pooja Singh | Associate Professor, Department of Computer Science and Engineering, of School of Computing Science and Engineering, Galgotias University, Greater Noida.

The matter presented in the project has not been submitted by us for the award of any other degree of this or any other places.

Sarthak Luthra

(19SCSE1180075)

Shahreen Ali

(20SCSE1180053)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Dr. Pooja Singh

Associate Professor

CERTIFICATE

The Review-1 Project Viva-Voce examination of Sarthak Luthra (19SCSE1180075) and Shahreen Ali (20SCSE1180053) has been held on 28th March, 2023 (Monday) and his/her work is recommended for the award of B. Tech. Computer Science and Engineering in specialization with Artificial Intelligence and Machine Learning.

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: 28 March, 2023

Place: Greater Noida

Abstract

Sign language recognition is a critical problem in computer vision and machine learning, with the potential to improve communication and accessibility for deaf and hard-of-hearing individuals. In this research paper, we propose a novel approach for sign language recognition from the live feed using Tensorflow.js, a JavaScript library for machine learning in the browser.

Our approach involves the use of convolutional neural networks to extract features from the video sequence, and a pre-trained model known as MobileNetV2 to classify images correctly. We have also introduced the autocorrect feature in order to make real-time detection faster. Our final model runs at 15FPS and detects finger spellings with an accuracy of 90%. Our model takes into account both the temporal and spatial features of the video sequence.

Our results suggest that our approach has the potential to be a powerful and effective tool for sign language recognition, with the advantage of being a Next.js application which runs entirely in the browser and creating a PWA version of the app we make it accessible to a wider audience. Overall, our research represents a significant step towards improving the accessibility and inclusivity of sign language recognition.

Keywords — Tensorflow.js, Next.js, PWA, convolutional neural network, MobileNetV2, temporal and spatial features.

Table of Contents

List of Figures

Acronyms

CNN	Convolution Neural Network
RNN	Recurrent Neural Network
PWA	Progressive Web Application
ASL	American Sign Language
SVM	Support Vector Machines
ISL	Indian Sign Language

CHAPTER-1

INTRODUCTION

Sign Language is an important means of communication for deaf and hard-of-hearing individuals, enabling them to express themselves and interact with others in a meaningful way. However, sign language interpretation remains a challenge, required specialized training and expertise. In recent years, advances in computer vision and machine learning have led to the development of sign language recognition systems, which can automatically detect and interpret sign language gestures. By synthesizing the findings of previous studies, this research paper aims to contribute to a better understanding of the current state-of-the-art in sign language gesture recognition and to inform the development of more accurate, reliable and user-friendly application for interpreting sign language. Further, translating the interpreted gestures into text or speech in real-time.

1.1. SIGN LANGUAGE

Sign Language is a visual language that uses hand gestures, facial expressions, and body movements to communicate meaning.

It is used by deaf and hard-of-hearing individuals as their primary mode of communication, but can also be used by individuals with normal hearing who want to communicate with the deaf community. There are many different sign languages used around the world, with their own unique grammar, syntax, and vocabulary. For example, American Sign Language (ASL) is used

primarily in the United States and Canada, while British Sign Language (BSL) is used in the United Kingdom.

Sign Language is a rich and complex language, with the ability to convey a wide range of emotions, ideas and concepts. It is not simply a direct translation of spoken language, but has its own unique rules and structure. Sign language can be learned and used by anyone, regardless of their hearing ability, and is an important tool for promoting inclusivity and accessibility for the deaf community.

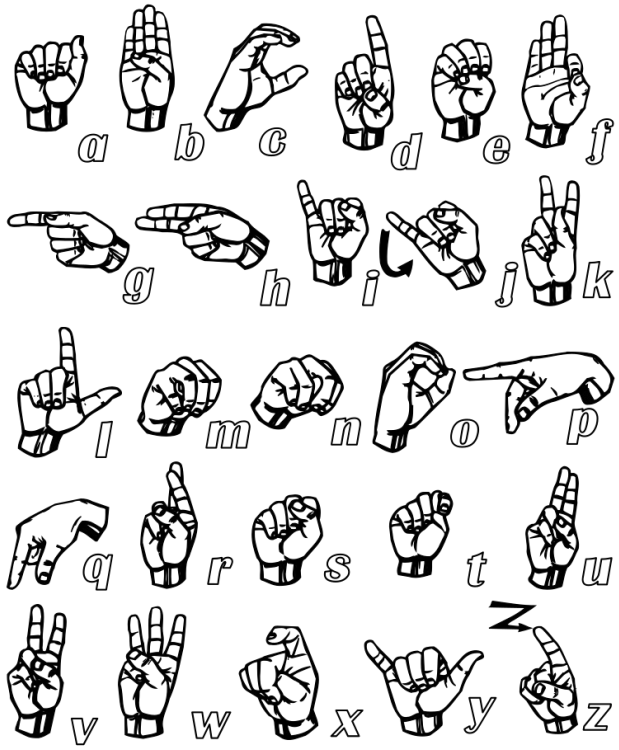

Fig 1. Single hand Sign Language for alphabets

Sign language is a visual language and consists of 3 major components:

- Fingerspelling: Used to spell words letter by letter.
- Word level sign vocabulary: Used for the majority of communication.
- Non-manual features: Facial expressions and tongue, mouth and body position.

CHAPTER-2

LITERATURE SURVEY

Singha, Joyeeta and Karen Das's [1] research paper titled "Automatic Indian Sign Language Recognition for continuous Video Sequence" presents an approach for the automatic recognition of Indian Sign Language (ISL) gestures from continuous video sequences. The authors use a combination of image processing techniques and machine learning algorithms to recognize ISL gestures. The four major modules in this paper are Data Acquisition, Pre-processing, Feature extraction and Classification. The authors first segment the video sequences into individual frames and then apply image processing techniques to extract hand regions from each frame. They then use a feature extraction algorithm to generate a feature vector for each hand region, which is used as input to a machine learning algorithm for recognition. The machine learning algorithm used in this paper is the k-nearest neighbor (k-NN) algorithm, which is a simple and effective classification method. The authors evaluate their approach on a dataset of 700 ISL gesture sequences performed by five different signers. The results shows that their approach achieves an average recognition accuracy of 88.7%.

A. Bhat, V. Yadav, V. Dargan, and Yash's [2] research paper titled "Sign Language to Text Conversion using Deep Learning" is an important contribution to sign language recognition and translation. The paper presents an approach for converting ISL gestures to text using deep learning techniques. The authors use a convolutional neural network (CNN) to recognize ISL gestures from video input, and a recurrent neural network to convert recognized gestures to text. The authors use a dataset of 1250 ISL gestures performed by ten different signers for training

and evaluation. The results show that the proposed approach achieves an accuracy of 93.8% in recognizing ISL gestures, and a word-level accuracy of 85.2% in converting the recognized gestures to text. The use of large dataset and evaluation of the approach on multiple signers increase the generalizability of the results. However, the approach is evaluated on a limited vocabulary of 100 words, which may not be sufficient for practical application.

The research paper [3] titled “Gesture and sign language recognition with temporal residual networks” by L. Pigou, M. Van Herreweghe, and J. Dambre, focuses on an approach for recognizing gestures and sign language from video sequences using temporal residual networks. The authors use a deep learning architecture consisting of 3D convolutional layers and residual blocks to learn spatiotemporal features from the video sequences. The authors evaluate their approach on two datasets, one for recognizing isolated gestures and the other for recognizing continuous sign language. The result show that the proposed approach outperforms the state-of-the-art methods on both datasets, achieving an accuracy of 98.6% and 91.8% respectively.

The research paper by Nandy, Anup & Prasad, Jay & Mondal, Soumik, Chakarborty, Pavan & Nandi, G., titled “Recognition of Isolated Indian Sign Language Gesture in Real time” [4], presents an approach to recognize Isolated ISL in real-time. The authors use an optical sensor-based system to capture the video data of ISL gestures, and then extract the relevant features using the discrete cosine transform (DCT).

The authors use support vector machines (SVMs) to classify the gestures based on the extracted features. The authors evaluate their approach on a dataset of 30 ISL gestures performed by six

signers. The results show that the proposed approach achieves an accuracy of 96.67% in recognizing the ISL gestures in real-time. The use of SVMs for gesture recognition, along with the use of DCT for feature extraction, shows the effectiveness of these techniques for recognizing ISL gestures. The real-time nature of the approach makes it suitable for practical applications.

Research paper titled “SignText: a web-based tool for providing accessible textbook contents for Deaf learners” by Joy, J., Balakrishnan, K. and Madhavankutty, S. [5], presents a web-based tool for providing accessible textbook content for deaf learners. The authors present SignText, a tool that enables the creation of digital textbooks with sign language videos and text in multiple languages. The tool allows educators to create custom content for deaf learners and can be used in conjunction with existing textbooks. The authors evaluate SignText through a user study with deaf learners and educators. The results show that it is easy to use and improves the accessibility of textbook content for deaf learners. The scalability of the approach by creating textbooks for multiple subjects and languages is also demonstrated.

The research paper titled “P2P audio and video calling application using WebRTC” by Majid, H., A Samah, A., Mi Yusuf, L., Nasien, D., and Cheah, T.L. [6], presents a peer-to-peer audio and video calling application developed using WebRTC. WebRTC (Web Real-Time Communication) is an open-source technology that enables real-time communication capabilities, such as video and audio streaming, directly in web browsers. The authors leverage the WebRTC technology to develop a P2P audio and video calling application that enable users to communicate in real-time without the need for third-party software. The authors evaluate the

performance of the application through a series of experiments including the application in different network conditions and evaluating the application's performance in terms of latency, jitter and packet loss. The results show that the application performs well under different network conditions and provides low-latency and high-quality audio and video streams.

"Word-level Deep Sign Language Recognition from video: A New Large-scale Dataset and Methods Comparison" by Li, D., Opazo, C. R., Yu X., and Li, H. [7], created a new large-scale dataset and methods comparison for word-level deep sign language recognition from video. The authors first introduce a new dataset called the Chinese Sign Language Recognition Dataset (CSLRD), which consists of 3000 video clips and covers 500 Chinese Sign Language words. The authors then compare various deep learning-based methods for word-level sign language recognition using the CSLRD dataset. The methods compared include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and a combination of CNNs and RNNs. The authors evaluate the performance of the different methods using several metrics, including recognition accuracy, recognition speed and memory usage. The results show that the combination of CNNs and RNNs outperforms the other methods in terns of accuracy, while CNNs are the fastest and require the least amount of memory. The authors also conduct a comparative analysis of their dataset with other sign language recognition datasets, and the results indicate that their dataset covers a more diverse set of sign language words and has a higher number of samples per word.

CHAPTER-3

WORKING OF PROJECT

3.1. ALGORITHMS

3.1.1. TENSOFLOW.JS

TensorFlow.js is a JavaScript library for training and deploying machine learning models in the browser or on Node.js. It is based on the TensorFlow library, which is a popular open-source machine learning framework. With TensorFlow.js, developers can build and train machine learning models directly in the browser using JavaScript, without the need for additional backend infrastructure or server-side code. TensorFlow.js provides APIs for building and training models, as well as for loading pre-trained models and running inference. It also supports running models on multiple devices, including CPUs, GPUs, and mobile devices.

3.1.2. NEXT.JS

Next.js is a popular open-source React framework for building server-side rendered (SSR) web application. It provides a set of powerful features such as automatic code splitting, server-side rendering and static site generation. One key benefit of using Next.js is that it enables developers to create fast and SEO-friendly web applications without having to worry about the underlying infrastructure. The framework automatically

optimizes code and resources to ensure fast load times and smooth user experiences.

Next.js also includes several other features, such as support for CSS-in-JS, hot module replacement, and dynamic imports, which make it an ideal choice for building modern web applications.

3.1.3. WEBSOCKET AND WEBRTC

WebSockets and WebRTC are two popular web technologies used to facilitate real-time communication and data transfer between clients and servers over the internet. WebSockets provide a bi-directional, persistent communication channel between a client and a server, allowing data to be transmitted in real-time without the need for repeated HTTP requests.

WebRTC on the other hand, is a peer-to-peer technology that enables real-time audio, video and data sharing between web browsers without the need for native applications.

We have used these two powerful technologies to add a live video communication functionality in our real-time sign language recognition Next.js application [6]. This will provide a seamless experience to anyone who wants to communicate with the deaf community without learning the sign language.

3.1.4. CONVOLUTION NEURAL NETWORK (CNN)

A Convolutional Neural Network (CNN) is a type of feed-forward deep learning algorithm that is primarily used for image classification and object detection tasks. It is based on the idea that local patterns in an image are important for identifying the overall object.

A CNN consists of multiple layers, including convolutional layers, pooling layers, and fully connected layers. In the convolutional layer, the input image is passed through a series of filters, which extract features from the image. The pooling layer is used to downsample the feature maps produced by the convolutional layer, reducing the dimensionality of the data. Finally, in the fully connected layer, the feature maps are flattened and fed through a traditional artificial neural network for classification.

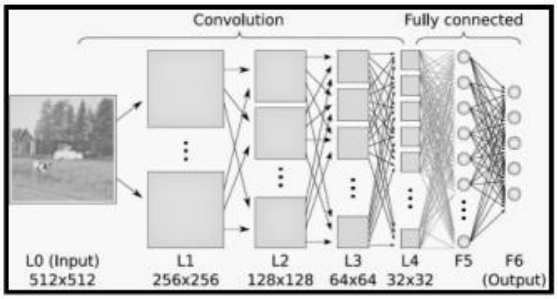

Fig 2. CNN

There are four main steps in CNN: Convolution, Subsampling, Activation, Full connectedness.

3.1.5. RECURRENT NEURAL NETWORK (RNN)

RNN stands for Recurrent Neural Network, which is a type of neural network that is used for processing sequential data. It is an artificial neural network that has a feedback loop between the hidden layers, allowing it to process inputs of variable length and size. The output of each layer is fed back as input to the next layer, allowing the network to maintain a "memory" of previous inputs. This makes it particularly effective for processing time-series data, such as speech recognition, handwriting recognition, and natural language processing.

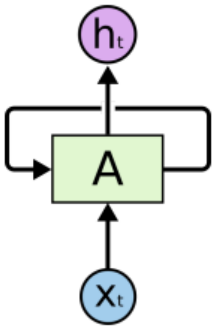

Fig 3. A chunk of Recurrent Neural Network

In RNNs, each neuron receives input not only from the previous layer but also from the previous time step. This allows the network to capture temporal dependencies between inputs, which is important in many applications. The hidden state of the network is updated at each time step, allowing the network to learn from the previous inputs and modify its current state accordingly.

One of the key advantages of RNNs is their ability to process variable-length sequences of data, which makes them useful in a wide range of applications.

3.2. IMPLEMENTATION

The app HandWave is video conferencing application for deaf community which uses machine learning to recognize the finger spelling or sign language gestures and convert them into text and further text to speech. It is built using following technologies:

1. Python
2. Tensorflow.js
3. OpenCV
4. HTML/CSS
5. Next.js
6. WebRTC and WebSocket

Our application bridges the gap faced by the deaf community, who primarily use American Sign Language (ASL) to communicate. Once it is out in the market, it has the potential to make a positive impact and provide of pleasing experience to nearly 6 million deaf individuals in India.

The complete working of the project is as follows:

1. First the user opens HandWave PWA or search for it in their browser.
2. A home screen where they can start a meeting by just filling in some personal details or simply login using google.
3. The app will ask for camera permission, once granted. Our machine learning model starts predicting the letter being signed in each frame of the canvas region.
4. The predicted characters are filtered using an algorithm to extract the corresponding text.
5. In the event that the model misclassifies or overlooks a letter, the autocorrect feature is present to rectify any misspelled words.
6. Once the user has completed signing their text, the app utilizes text-to-speech technology to read the text aloud, thereby providing them with a more seamless video calling experience.
7. We have also created a Progressive Web App version of our application to make it readily accessible on there mobile home screen just like other native application.

This application is mobile responsive meaning, it is compatible to devices of all screen sizes and comes with simple User Interface which makes it accessible to a broader age group of users.

3.3. SYSTEM ARCHITECTURE

OpenCV pipeline to transform an image to a letter output.

cv.cvtColor(img, result, cv.COLOR_BGR2GRAY);

cv.adaptiveThreshold(result, result, 255, cv.ADAPTIVE_THRESH_GAUSSIAN_C, cv.THRESH_BINARY, 21, 2);

cv.cvtColor(result, result, cv.COLOR_GRAY2RGB);

To facilitate the Convolutional Neural Network's ability to "learn" how to classify frames into letters, we utilize an OpenCV edge-detection transformation to provide the network with a simplified image from which to extract information.

```
Downloading data from https://storage.googleapis.com/tensorflow/keras
9412608/9406464 [==============================] - 0s 0us/step
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
rescaling (Rescaling)        (None, 224, 224, 3)       0
_________________________________________________________________
mobilenetv2_1.00_224 (Functi (None, 1280)              2257984
_________________________________________________________________
dense (Dense)                (None, 128)               163968
_________________________________________________________________
dense_1 (Dense)              (None, 64)                8256
_________________________________________________________________
dense_2 (Dense)              (None, 28)                1820
=================================================================
Total params: 2,432,028
Trainable params: 174,044
Non-trainable params: 2,257,984
```

Fig 4. Model Architecture

We conducted several experiments with different model architectures and utilized transfer learning from various base convolutional neural networks, such as VGG19, AlexNet, and others. Eventually, we arrived at the architecture displayed here, which enables real-time model execution while maintaining a reasonably high level of accuracy. To achieve this, we leveraged the output of a pre-trained convolutional neural network (MobileNetV2) for providing the model with richer "prior knowledge," we increase its ability to classify images accurately and incorporated three dense layers to modify the output for hand sign classification purposes.

In order to achieve a balance between accuracy and speed, we integrated an autocorrect feature that enables the model to overlook certain letters while still producing the correct text. This approach has resulted in a good overall user experience, with the model running at 15 frames per second on mobile phones.

CHAPTER-4

RESULTS AND DISCUSSION

The application is functioning according to our proposed plan, and our tests have demonstrated accurate finger spelling predictions. The app performs best with a plain background, and in the absence of any object in the canvas region, it displays a "nothing" text.

Below are two screenshots that showcase the application's performance and user interface.

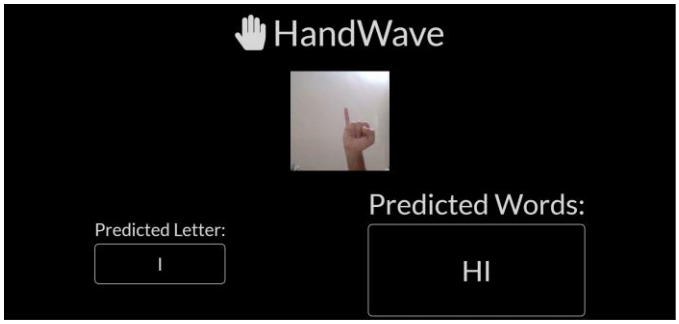

Fig 5. Predicting greeting word “HI”

Once the user has finished providing input, the text-to-speech function is activated, and the predicted word is pronounced aloud.

Fig 6. Predicting a reaction “OHHH”

CHAPTER-5

CONCLUSION AND FUTURE SCOPE

5.1. CONCLUSION

Sign language recognition is an important and challenging problem, with the potential to improve communication and accessibility for deaf and hard-of-hearing individuals. We have found that sign language recognition has made significant progress in recent years. Our application ‘HandWave’ bridges up the gap between normal people and deaf community and provides them a web-based platform for smooth communication.

However, there are several challenges and limitations in sign language recognition that need to be addressed, including the variability of sign language gestures, lack of dataset, and difficulty of interpreting he linguistic and cultural context of sign language. Further research in several areas, including the development of more sophisticated deep learning algorithms, the integration of context and language models and the exploration of multi-modal recognition systems that combine video and audio data, may solve these challenges.

5.2. FUTURE SCOPE

There’s always scope for improvements. We wish to improve the accuracy and robustness of sign language recognition system we proposed. We would like to make an upgraded version of our model to recognize words in ASL, as it will make the communication faster and easier for the users. We would like to directly get in touch with the individuals using this application so to get their feedback and fine-tune the application. Also, collaborating with the

deaf community to promote the app and make it accessible to target audience. Further we can add more features like, scheduling a call for later, real-time chat during the meeting would be helpful.

CHAPTER-6

REFERENCES

[1] Singha, Joyeeta, and Karen Das. "Automatic Indian Sign Language Recognition for Continuous Video Sequence." ADBU Journal of Engineering Technology 2, no. 1 (2015).

[2] A. Bhat, V. Yadav, V. Dargan and Yash, "Sign Language to Text Conversion using Deep Learning," 2022 3rd International Conference for Emerging Technology (INCET), Belgaum, India, 2022, pp. 1-7, doi: 10.1109/INCET54531.2022.9824885.

[3] L. Pigou, M. Van Herreweghe, and J. Dambre, “Gesture and sign language recognition with temporal residual networks,” in 2017 IEEE INTERNATIONAL CONFERENCE ON COMPUTER VISION WORKSHOPS (ICCVW 2017), 2017, pp. 3086–3093.

[4] Nandy, Anup & Prasad, Jay & Mondal, Soumik & Chakraborty, Pavan & Nandi, G.. (2010). Recognition of Isolated Indian Sign Language Gesture in Real Time. 70. 102-107. 10.1007/978-3-642-12214-9_18.

[5] Joy, J., Balakrishnan, K. & Madhavankutty, S. SignText: a web-based tool for providing accessible text book contents for Deaf learners. Univ Access Inf Soc (2021). https://doi.org/10.1007/s10209- 021-00801-7

[6] Majid, Hairudin & A Samah, Azurah & Mi Yusuf, Lizawati & Nasien, Dewi & Cheah, T.L.. (2016). P2P audio and video calling application using WebRTC. 11. 1766-1770.

[7] Li, D., Opazo, C. R., Yu, X., & Li, H. (2019). Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. ArXiv. /abs/1910.11006

[8] F. Ronchetti, F. Quiroga, C. A. Estrebou, and L. C. Lanzarini, “Handshape recognition for Argentinian Sign Language using ProbSom”, JCS&T, vol. 16, no. 01, pp. p. 1–5, Apr. 2016.