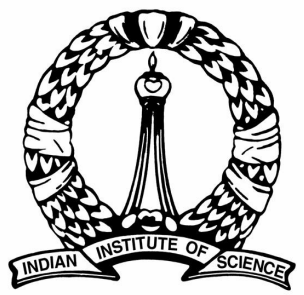
Indian Sign Language Recognition and Sentence Generation

A project report
submitted in partial fulfilment of the
requirements for the degree of
Master of Technology
in
Artificial Intelligence
by

MANGAL DEEP SINGH
SR No.: 04-03-06-10-51-21-1-19857

Under the guidance of

Dr. RATHNA G N



Electrical Engineering
Indian Institute of Science
Bengaluru - 560012
June 2023

Acknowledgements

First of all, I would like to express my gratitude to my advisor, Dr. Rathna G N for her guidance, encouragement, advice, supervision, and patience throughout the project work. I am also grateful to all the faculty members of the department of Electrical Engineering for their unparalleled teaching and academic support.

I thank my labmates Manish Aradwad, Viltu Jujhajiya and Vedpal Jangir for their support and suggestions.

Finally, I am deeply indebted to my parents and friends for their constant love and motivation.

Abstract

Indian Sign Language is used by the deaf and mute people in India for communication with each other through signs. However, there has been a communication gap between the deaf and mute people and the rest of the population. ISL lacks good resources for data driven tasks such as deep learning models which can be used for developing tools for sign language recognition.

During this project, I aimed towards narrowing down the communication gap and facilitate the communication.I have created a dataset by capturing gestures of some of the commonly used words and built a deep learning which can detect the gesture for a word correctly. Then, we have used the recognized words for sentence generation. We are translating from sign language to corresponding text.

The model works with webcam as well as any video file. So, it can be used for real-time translation facilitating communication between speech impaired group and the rest.

Abbreviations

LSTM	Long Short Term Memory
ReLU	Rectified Linear Unit
BERT	Bidirectional Encoder Representations from Transformers
ISL	Indian Sign Language
GSL	German Sign Language
ASL	American Sign Language
LLM	Large Language Model
CNN	Convolutional Neural Networks
BLEU	Bilingual Evaluation Understudy
HMM	Hidden Markov Model
KNN	K-Nearest Neighbor

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Problem and its importance

With the advancement of deep learning models, we have seem promising improvements in language processing tasks such as classification, translation and generation. These models required good size of data and ISL suffers from not having a large dataset. While much study has been done on the recognition of ASL, ISL varies greatly from ASL. The lack of dataset and the fact that there is no standard benchmark for ISL, and the sign language varies depending on the location, there has been limited work on ISL.

This projects aims to use ISL to bridge the communication gap between the speech impaired group of people and the rest. ISL is referred to the language used by the speech impaired people of India. As per the 2011 Indian Census, there are about 6 million deaf people in India. ISL is one of the most widely used sign language in the world. This has been a communication bar between the speech impaired community and the rest because of lack of awareness, limited work on ISL and lack of standard benchmarks.

The deaf and mute people communicate with each other by making gestures with the help of the hands mostly and sometimes with the help of other body parts too such as face gestures as well as pose. The motivation behind working on this project is to build a model using the advanced Deep Learning tools available to make it possible for the speech impaired and the general population to communicate with each other. With the help of technology available, we can narrow down the communication gap between the speech impaired community and the rest of the population. .

The idea is to precisely detect the gesture from the input and then use the recognized words to generate a meaningful text. This is done by recognizing the gestures of the person, then mapping it to its correct word and then using all the words, goal is to form a meaningful sentence which is being conveyed in the video. So, given a video a person performing some gestures, the model discussed in this work should be able to recognize the words for which the gestures were made and then form a sentence which is being conveyed in the video.

Chapter 2

Related Work

In the field of ISL Recognition, many attempts have been made and work has been done for recognising the gesture from videos and images using various methods and models depending on the sign language and the signs.

In the work done by Jing-hao Sun[12], they separated the hand, and CamShift algorithm was used to detect real-time hand gestures. Then, using CNN, the hand movement region was recognised and then used to classification of digits. The proposed system has dataset of total 1600 pictures for training dataset, 4000 hand gesture, 400 images for each type. This experiment shows accuracy about 98.3 percent.

J. Singha et al. [5] proposed a method for real time recognition where Eigen value-weighted Euclidean distance was used to classify signs. P. Kishore et al. [16] proposed a system by finding active contours from boundary edge map using Artificial Neural Network (ANN) to classify the signs. Another approach used the Viola Jones algorithm with LBP functions for hand gesture recognition in a real-time environment [21]. It had the advantage of requiring less processing capacity to detect the movements.

Most of these works were based on pattern recognition, feature extraction, and so on. However, in most of the cases, a system with single feature is not enough. Therefore, hybrid approaches were introduced to solve this problem. For eg, A. Nandy et al. [17] used hybrid approaches with KNN and Euclidean distance to classify gestures from orientated histogram features. The limitation of this approach was the poor performance in case of similar gestures. Hasan[13] used scaled normalisation to recognise gestures using brightness factor matching. Noor Tubaiz [14] proposed

using the k-Nearest Neighbor approach to classify sequential data. Data gloves are used to detect hand movements.To supplement the raw data, window-based statistical features are calculated from previous raw feature vectors and future raw feature vectors. To recognise terms in ISL, the proposed framework was developed using novel techniques based on existing systems. B. Bauer et al. described an approach for a continuous sign language recognition method. It is a framework that depends on continuous HMM images. It employs GSL. Feature vectors that represent manual signs are fed into the device.

However, for real-time systems, researchers needed a faster way to solve this problem. The advancements in Deep Learning technologies have enabled automation of image recognition using various image recognition models. G. Jayadeep et al. [19] used a CNN to extract image features, LSTM to classify these gestures and translate them into text. Bin et al. [20] proposed the InceptionV3 model to use depth sensors to identify static signs. It eliminated the steps of gesture segmentation and feature extraction.

Earlier work done by Pratik et al.[3], focused on translating a video sequence directly to text. Since, there is not much of sentence-level ISL data available, not much progress has been done in this direction.

Chapter 3

Method

This project is done in two stages. First step is to recognize the word from the gesture and then generate a sentence.

3.1 ISL Recognition

The first step is to recognize the gesture performed by the person. This can be done with the help of CNN as well as MediaPipe Holistic model. The problem with CNN is it requires very large data as compared to MediaPipe Holistic model to perform the recognition part. Also, the CNN model are large compared to the models used with MediaPipe model. So, I have used MediaPipe Holistic model which performs the gesture recognition part by detecting the landmarks of face, pose and both the hands to create a complete landmark of the human body as shown in Fig. 3.1. For each frame, it detects the landmark points and stores those values in a .npy file which is a NumPy array. This has been used to create the dataset. Landmark points for hands are shown in Fig. 3.2 and Fig. 3.3.

The next step is classification part where the model has to classify the input gestures as one of the words in the vocabulary. Since, the input data is sequential where the gestures performed depends on the other also. Since, the data is sequential, we have used LSTM which performs well with the sequential data. The MediaPipe model detects the gestures, feed it to the classifier, which gives a probability distrubution over the words in the vocabulary and the word with the

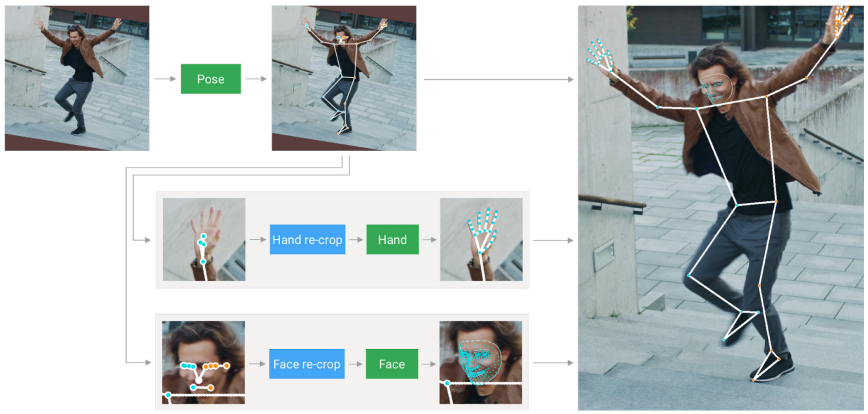
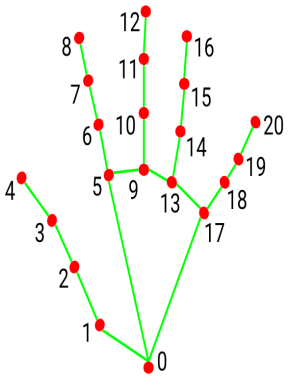


Fig. 3.1 MediaPipe Holistic Model.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

Fig. 3.2 MediaPipe 21 hand landmark points.

highest probability is returned as output. These recognized words are stored in a list. The LSTM architecture is shown in the following Fig. 3.4.

3.2 ISL Sentence Generation

The last step is to convert the recognized words into a sentence. This task has been done with the help of LLMs. For our task, the LLMs are fed the recognized words, which then generates a sentence using those words. A good LLM should be able to form meaningful sentences, which also convey the same meaning as in the video.

The data we are working with is a sequential data where the points in the dataset are dependent on other points. During inference, the recognized word depends on the sequence in which the actions

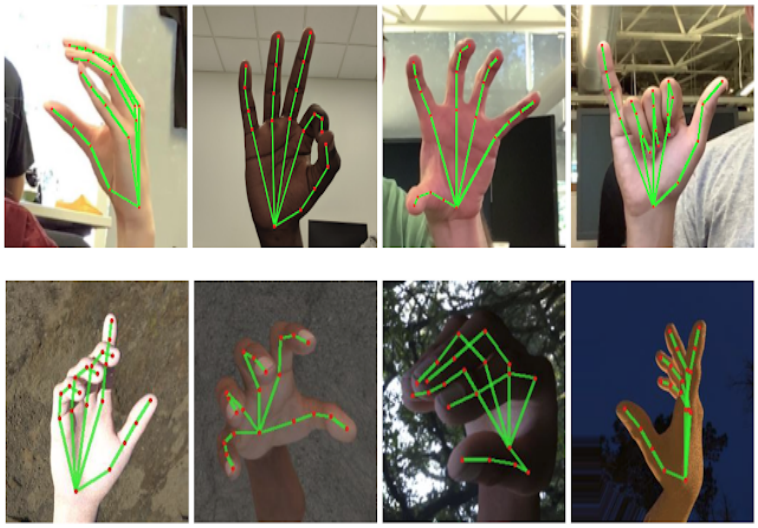


Fig. 3.3 MediaPipe detecting hand landmarks.

were performed. The model stores the 4 most recent detected words and then it is passed to BERT and ChatGPT LLM.

- **BERT:** In case of BERT, we have to give input along with [MASK] token, and it performs prediction for those masked words with the help of words around it. The position of the [MASK] token as well as the words matter a lot as they do not change, and may lead to meaningless sentences.
  - **Input:** [MASK] brush [MASK] you [MASK] eat [MASK] time
  - **Output:** a brush and you can eat next time.
- **ChatGPT:** We give a prompt as shown below and it returns a sentence using those words.
  - **Input:** Generate a small sentence using only the following words: brush, you, eat, time.
  - **Output:** You brush your teeth every time you eat.

ChatGPT is flexible, and the position of the words don't matter much as it is able to handle it well. Also, in the input, if there is repetition of words, then ChatGPT handles it in a much better way as compared to BERT.

In the previous work, the points captured for 30 frames for a sample were averaged and it became a single data point was assigned to each frame. In this task, we have leveraged the availability of the sequential data available. So, the model can perform better during training as well as inference.

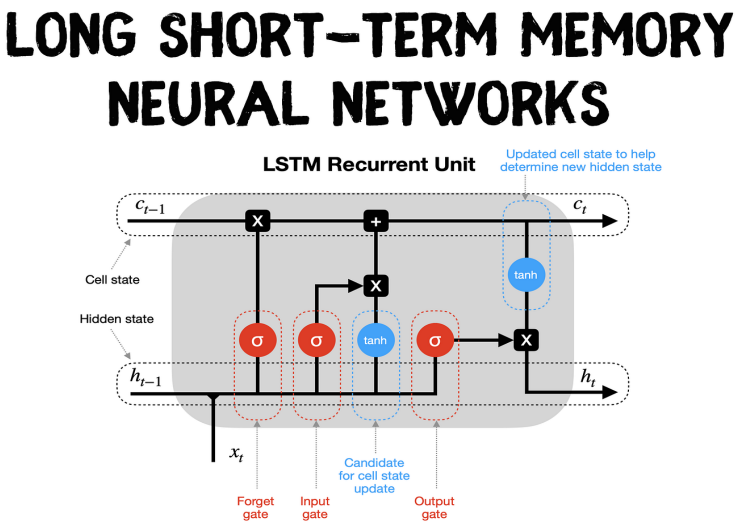


Fig. 3.4 LSTM Architecture

As this is a classification task, I have used LSTM architecture which is fit for our purpose. For each sample, 30 frames of data is passed in a sequential order instead of a single frame data.

Chapter 4

Experimental Details

4.1 Dataset

There is not enough available dataset for training a deep learning model and have good results. The ISL dataset available on the ISLRTC YouTube channel is not large enough to train deep learning models because each word has only one or two gesture which is not sufficient. So, we first need a good and large dataset.

In the previous work, the training was done on the American Sign Language and the inference was done on the ISL. But there is a basic difference between the two. ASL uses mostly one hand for the gestures whereas ISL uses both the hands. Also, for many alphabets and words, the gesture is very different is the two sign languages. So, it might work for the gestures common in both but not for the different ones. So, it is not ideal to train the model on ASL and do the inference on ISL data.

For our task, I have dedicated some time on creating a working dataset. To make the model robust, I have created the dataset in varying conditions like light, distance, person and angle. I created the dataset for 30 words using the MediaPipe Holistic model. For each word, I have recorded 30 sample videos and for each sample video, the model captures 30 frames and coverts it to a .npy files containing 258 points. I have not stored the Face landmarks as they do not contribute a lot to our task. In fact, it can badly affect the model as Face has a total of 1404 points out of 1662 and Face points are almost same during each gesture. Fig. 4.1 and 4.2 shows the landmark points with and without Face landmark respectively.

Total number of words	30
Number of samples for each word	30
Total landmark point in a sample	258
Dataset size	(30, 30, 258)

Table 4.1 ISL Dataset.

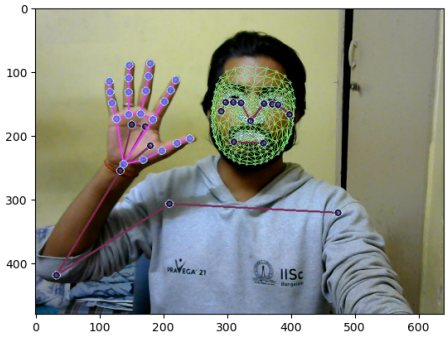


Fig. 4.1 With Face Landmark.

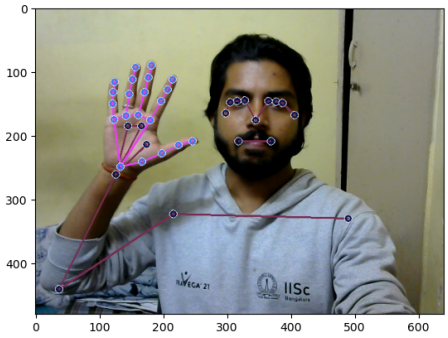


Fig. 4.2 Without Face Landmark.

1. I have created the dataset by capturing the landmarks of the gestures from the video. The model is set to capture 30 frames per second (30 fps). For each frame, the MediaPipe holistic model detects the landmarks and store it as a .npy file which is a NumPy array file.

2. Now, for Face landmarks, it stores a total of 1404 points i.e., 468*3 points where 3 represents the RGB components, and 468 represents the total number of landmarks.

3. For Pose estimation, it stores a total of 132 (=33*4) points where 4 represents the RGB and the visibility component, and a total of 33 landmarks.

4. For Left and Right hand landmarks, it stores 63 (=21*3) points for each hand and 21 landmarks for each hand.

4.2 Train-Test Split

For the training of our model, I have used 80% of the total data. Also during training, 20% of the test data has been used for validation. And the remaining of the data has been used for testing part.

Parameters	Value
Total number of layers	6
Activation function	relu
Number of epoch	200
Input shape	(30, 258)
Optimizer	Adam
Loss function	categorical_crossentropy
Evaluation metrics	categorical_accuracy

Table 4.2 Parameters of LSTM model.

Epoch	Accuracy
20	0.15
50	0.48
80	0.76
100	0.74
150	0.82
180	0.94
200	0.90

Table 4.3 Training accuracy vs Epoch.

4.3 Training

The LSTM model has been trained as follows:

- In our dataset, there are 120 words and each words has 30 video samples collected. Each sample has 30 .npy files corresponding to the 30 frames taken in sequential order storing the landmarks detected.
- To each sample, a label is appended and then it is fed to the LSTM model for training as a classification task.

The loss vs epoch graph is shown in Fig. 4.3.

The accuracy vs epoch graph is shown in Fig. 4.4.

From the Fig. 4.4, we can see that train accuracy approaches to 1 after training for 200 epoch. The training accuracy at different epoch is shown in table 4.3.

4.4 Evaluation Metric

For the LSTM classification model, the evaluation metric used is categorical accuracy. Categorical accuracy calculates the percentage of predicted values that match the actual values for one-hot labels. It identifies the index at which the maximum value occurs and if it is same for both predicted and actual value, then it is considered accurate. From the Fig. 4.4, we can see that train and validation accuracy reaches almost 1 during training for 200 epoch. The accuracy of the model on test data is about 95%.

The second model is the ChatGPT LLM where the output of the model can be used for evaluation against the actual sentence. Since not enough sentence-level data is available, we have compared the outputs of BERT and ChatGPT for simple sentences. From table 5.1, we can see the outputs of BERT and ChatGPT for the same input. We can clearly see that ChatGPT is a better model for our task.

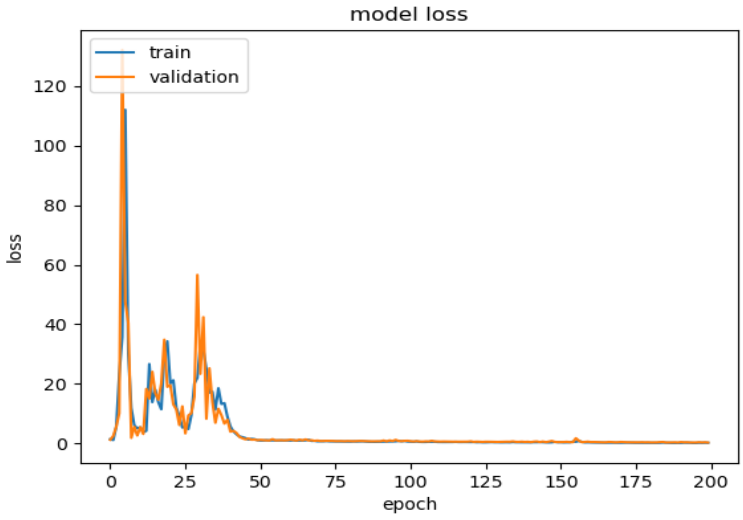


Fig. 4.3 Loss vs Epoch Graph

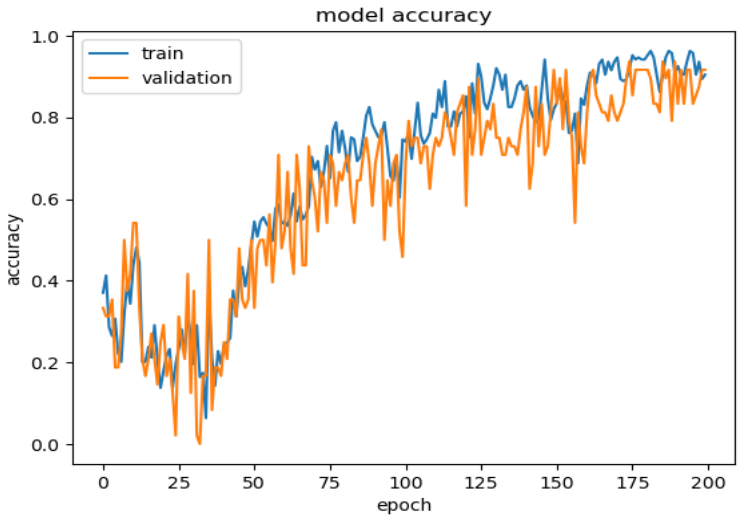


Fig. 4.4 Accuracy vs Epoch Graph

Chapter 5

Results

I have tested the model for word-level recognition. The input can be given using webcam as well as video file. The model detects the gestures shown in the frame and converts it to .npy file which is them sent to the LSTM for classification. The LSTM model takes last 30 frames and classifies based on the probability score from SoftMax layer.

I have used two key values: threshold and number of words stored. During prediction, if the confidence of the model is above the threshold value, then it classifies the gesture or else wait for more frames for detection. The threshold value for our model is 0.5. Number of words passed to the LLM model for sentence generation can impact the quality of the sentence. I have stored the recent 4 words from detected words and used them for sentence generation using ChatGPT.

Figures below shows the model detecting the words from the gesture shown in the frame.

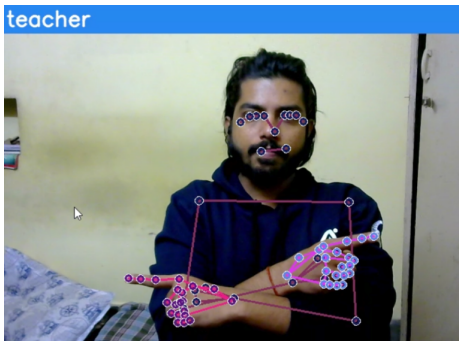


Fig. 5.1 Detecting word **teacher**.

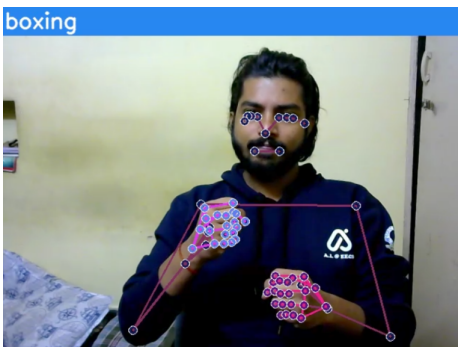


Fig. 5.2 Detecting word **boxing**.

Fig. 5.3 Model detecting the words and storing them.

Input words	**BERT Output**	**ChatGPT Output**
brush, you, eat, time	a brush and you can eat next time.	You brush your teeth every time you eat.
what, name, you	what a name are you?	What is your name?
I, time, you, boxing	I have time when you in boxing.	I have time for you to go boxing.

Table 5.1 Language model output for recognized words.

Chapter 6

Discussion

While working on this project, the aim was to help the speech impaired and the rest of the population communicate with each other and narrow the communication gap. The unavailability of enough ISL resources led me to working on creating a working dataset. ASL dataset is available but ASL uses single hand whereas ISL uses both hands. Also, due to diverse population of India, there are slight variation of gestures for a word depending on the geography. Also, there are demography specific words. The other problem is the order of gestures because sometimes order of the gestures is not fixed. All these are challenges that were faced while working on this project.

Chapter 7

Future Scope

The ISL Sentence Generation model is for recognising the gestures and translating them into text. This can be used for human-machine communication as well as narrow communication gap between speech impaired group and the rest.

Since all the models used in this project are state of the art such as MediaPipe Holistic model for gesture detection, LSTM for classification of sequential data and ChatGPT LLM for sentence generation, possible future work should be creating a large dataset for ISL by extending the corpus which can then be used to build ISL related models. Dataset for common phrases and words will help in building a more robust and working model for translation from gesture to text. Also sentence-level translations can be added to the dataset. Also, there is scope for creating a good sentence-level dataset for ISL.

After extending the corpus, a good ISL to text translation model can be built and can build an android and/or web application for the same.

Bibliography

[1] Neel Kamal Bhagat, Y. Vishnusai, and G. N. Rathna. Indian sign language gesture recognition using image processing and deep learning. In 2019 Digital Image Comput- ing: Techniques and Applications (DICTA), pages 1–8, 2019. doi: 10.1109/DICTA47822.2019.8945850.

[2] Bo Liao, Jing Li, Zhaojie Ju, and Gaoxiang Ouyang. Hand gesture recognition with generalized hough transform and dc-cnn using realsense. In 2018 Eighth International Conference on Information Science and Technology (ICIST), pages 84–90. IEEE, 2018.

[3] P. Likhar and G.N. Rathna. Indian sign language translation using deep learning. In In 2021 IEEE 9th Region 10 Humanitarian Technology Conference (R10-HTC) (pp. 1-4). IEEE., 2021.

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre- training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

[5] Joyeeta Singha and Karen Das. Indian sign language recognition using eigen value weighted euclidean distance based classification technique. arXiv preprint arXiv:1303.0634, 2013.

[6] Pratik Likhar, Neel Kamal Bhagat, and Rathna G N. Deep learning methods for indian sign language recognition. In 2020 IEEE 10th International Conference on Consumer Electronics (ICCE-Berlin), pages 1–6, 2020. doi: 10.1109/ICCE-Berlin50680.2020.9352194.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition, pages 248–255. Ieee.

[8] V. S. Kulkarni, S.D.Lokhande, (2010) "Appearance Based Recognition of American Sign Language Using Gesture Segmentation", International Journal on Computer Science and Engineering (IJCSE), Vol. 2(3), pp. 560-565.

[9] Geethu G Nath and Arun C S, "Real Time Sign Language Interpreter," 2017 International Conference on Electrical, Instrumentation, and Communication Engineering (ICEICE2017)

[10] Pigou, L., Dieleman, S., Kindermans, P.-J., Schrauwen, B. (2014). Sign Language Recognition Using Convolutional Neural Networks.

[11] CJ Sruthi and A Lijiya. Signet: A deep learning based indian sign language recognition system. In 2019 International conference on communication and signal processing (ICCSP), pages 0596–0600. IEEE, 2019.

[12] Jing-Hao Sun,Ting-Ting Ji, Shu-Bin Zhang, Jia-Kui Yang, Guang-Rong Ji "Research on the Hand Gesture Recognition Based on Deep Learning",07 February 2019

[13] Mokhtar M. Hasan, Pramoud K. Misra, (2011). "Brightness Factor Matching For Gesture Recognition System Using Scaled Normalization", International Journal of Computer Science Information Technology (IJCSIT), Vol. 3(2).

[14] Noor Tubaiz, Tamer Shanableh, and Khaled Assaleh, "Glove-Based Continuous Arabic Sign Language Recognition in User-Dependent Mode," IEEE Transactions on Human-Machine Systems, Vol. 45, NO. 4, August 2015

[15] B. Bauer,H. Hienz "Relevant features for video-based continuous sign language recognition", IEEE International Conference on Automatic Face and Gesture Recognition, 2002.

[16] P.V.V. Kishore, D.A. Kumar Optical flow hand tracking and active contour hand shape features for continuous sign language recognition with artificial neural networks IEEE 6th international conference on advanced Computing (2016)

[17] Anup Nandy, Jay Shankar Prasad, Soumik Mondal, Pavan Chakraborty, Gora Chand Nandi Recognition of isolated Indian sign language gestures in real time International Conference on Business Administration and Information Processing (2010)

[18] Shanmukha Swamy, M.P. Chethan, Mahantesh Gatwadi Indian sign language interpreter with android implementation Int J Comput Appl (2014), pp. 975-8887

[19] G. Jayadeep, N.V. Vishnupriya, V. Venugopal, S. Vishnu, M. Geetha Mudra: convolutional neural network based Indian sign language translator for banks 4th International Conference on Intelligent Computing and Control Systems (ICICCS), 2020 (2020), pp. 1228-1232

[20] B. Xie, X.y. He, Y. Li RGB-D static gesture recognition based on convolutional neural network J Eng, 2018 (16) (2018), pp. 1515-1520

[21] Hemina Bhavsar, Jeegar Trivedi Indian sign language recognition using framework of skin color detection, Viola- Jones algorithm, correlation-coefficient technique and distance based neuro-fuzzy classification approach Emerging Technology Trends in Electronics, Communication and Networking, 1214 (2020), pp. 235-243