Real Time Indian Sign Language Detection Using LSTM and Keypoint Extraction

Ezhil Tharsan S.
Department of Computer Science and Engineering
Sri Sivasubramaniya Nadar College of Engineering
Chennai,India

Dharshan S.
Department of Computer Science and Engineering
Sri Sivasubramaniya Nadar College of Engineering
Chennai,India

Dinesh G.
Department of Computer Science and Engineering
Sri Sivasubramaniya Nadar College of Engineering
Chennai,India

Saraswathi S.
Department of Computer Science and Engineering
Sri Sivasubramaniya Nadar College of Engineering
Chennai,India

ABSTRACT

Hearing and speech impaired people find it challenging to converse with ordinary. In India, about 63 million (6.3%) of the population suffers from auditory loss. They communicate through sign languages, which are based on visually conveyed sign patterns, which typically include hand movements. Sign language is not universal and not easy to learn. Many Artificial Intelligence (AI) based systems are present for the Sign Language conversion as a typical solution to this problem. But, most of the existing solutions, use American Standard Sign Language (ASL) . As per the research done for this project, Each country has its own sign language variant and in India, Indian Sign Language (ISL) is being followed. The major goal of the proposed system is to create an ISL Hand Gesture Motion Translation Tool that will assist the hearing and speech challenged community in transforming their voice and ideas into text so that they may communicate with others without huddles. The methodology is to identify isolated words and to design a dynamic hand gesture recognition system for ISL. The proposed system is built using Mediapipe which is used for key-point extraction combined with a Long Short-Term Memory (LSTM), a Recurrent Neural Network (RNN) architecture system design with Dense layers.It is capable to identify ISL Alphabets and words in real time environment.

General Terms

Sign Language, Indian Sign Language, Gesture Recognition, Facial Recognition, Recurrent Neural Network, Long Short-Term Memory, Dense Layer

Keywords

Keypoint Extraction, Dataset

1. INTRODUCTION

In India, about 63 million of the population suffers from auditory loss. They communicate through sign languages - ISL. The Indian Sign Language detection system's goal is to provide a strategy for translating sign language signs and gesture patterns into text. It would be feasible and have a great impact for the deaf-mute communities to complete any task involves communication with others without the assistance. Different sign languages are used in different regions or countries. The proposed system is built upon ISL as its medium of communication.

Sign Languages, which are visually transmitted sign patterns that generally include hand gestures, are used by hearing and speech impaired people to communicate. Sign languages are difficult to learn and non-universal, creating a communication gap between the hearing impaired and the general public. The motivation for developing such a useful application stemmed from the fact that it would aid in increasing social awareness and reducing these people's separation from the community.

Here, the proposed real time system will ease the communication between hearing-impaired and others, which converts gestures of ISL into equivalent Text. The designed system of ISL Hand Gesture Motion Translation Tool is built upon Long short-term memory (LSTM), Recurrent Neural Network (RNN) architecture with Mediapipe module for key-point extraction .

It is extremely difficult to create an automatic sign recognition system for ISL recognition. The main challenge was the lack of dataset availability for Indian Sign Language. Gestures were difficult and complicated. It is also more difficult in the case of ISL when compared to other sign languages because most signs require both hands and body posture. For some of the signs, the hand makes contact with the body. Background changes and real-time video angles are critical factors. Because ISL was only recently got popularly recognised and standardized, considering all of these as challenges in order to bring a better version of the hand gesture recognition system. This system can also be adapted to different environments based on needs and usage.

The following sections Related Work, Proposed System, Methodology, Test Result and Performance Evaluation were explained below.

2. RELATED WORK

In recent times, Sign Language and recognition is a wide area of research. Sign Language is the sole medium through which deaf-mute people can communicate their thoughts and feelings to others. The research work and survey done for this project about various sign language detection methodologies is discussed below.

The paper [9] proposed a LSTM Sign Language system with Handcrafted features and Hidden Markov Models which are used in traditional methods of Sign Language Recognition. However, dependable handcrafted features are difficult to design and are incapable of adapting to the wide range of sign words. To address this issue, Long Short Term Memory can finely model the contextual information of a temporal sequence. They used skeleton joint trajectories instead of the Kinect's colour and depth data. Four skeleton joints were fed into the LSTM neural network (left and right hands, left and right elbows). For evaluation, they created two Chinese isolated datasets: dataset I with 100 daily words and dataset II with 500 sign words. When the system's performance is compared to HMM, it achieves an 86%, 63% for datasets I and II.

The paper [10] proposes a novel hand gesture recognition scheme targeted to leap motion data. They used two sensors namely Leap motion and kinect for collecting the feature. Finally they combining all the various features from two sensors which gives the optimal accuracy of 91.3%. The system proposed in [4] Convolutional Neural Network algorithmic rule for the identification and categorization of the twenty six Indian language letters into their identical alphabet letters by capturing a time period image of that sign and changing it to its text equivalent. The GrabCut algorithm is used here for segmentation and MobileNet as image classification is used. The outcomes indicated a 96% precision for the testing images and a precision of 87.69% for runtime image.

The paper [5] proposes a approach for two way Indian Sign Language (ISL) communication.Their methodologies consist of two modules Sign-to-text(STT) and Text-to-sign(TTS).They used a Motion Frame Detection is to identify the frames and Frame Reduction algorithm is used to reduce the number of redundant frames to save processing time.ResNet algorithm from ImageAI is used for Pattern Matching.

In paper [14], For feature extraction, an algorithm is proposed for hand gesture detection in Indian Sign Language using four distinct features such as image restructuring, feature selection, edge detection, and rotation. For detecting hand gestures, they used the Discrete Cosine Transform, the Discrete Wavelet Transform (DWT), and an edge detection algorithm. In a real-time environment, it recognises Indian Sign Language numbers with nearly 87% accuracy. [11] provides 99.23% accuracy with the same work used for feature extraction and the K-Nearest Neighbors (KNN) classifier for sign recognition.

Recent work done with real-time background in accordance of paper [2], a deep learning methods has been adopted by convolution neural network(CNN) model to extract the sign language feature where for classification softmax layers is used. Both the simple and complex background alphabets have been considered. proposed architecture which achieved the accuracy of 99.10%,92.69%,and 95.95% for simple,complex and mixed background. The paper [15] presents a real-time two way system for communication between hearing impaired and normal people which converts the Indian sign language(ISL) letters into equivalent alphabet letters and vice versa. For the Feature extraction they used canny edge detection algorithm.

From the survey, it is been identified that many of the existing system mainly based on ASL. The discussed system in this project, built upon ISL using LSTM with Keypoint Extraction.

3. PROPOSED SYSTEM

In this proposed method, a vision-based categorization system for Indian Sign Language with dynamic signs (ISL). In contrast to a static sign, a dynamic sign motions comprises a complex motion of gestures with additional movements. A static sign is determined by a certain arrangement of the hands, whereas a dynamic sign is determined by a sequence of hand motions and configurations. The dataset is created from the scratch for the Indian sign language alphabets and common words. Mediapipe module is used for the keypoint extraction. The face and hand region are considered as the region of interest and mainly the keypoints are extracted from these regions. Then the keypoints extracted are given as an input to the built LSTM model and then the camera shown signs converted to the equivalent text form. The below Figure 1 represents some of the signs present in the dataset.

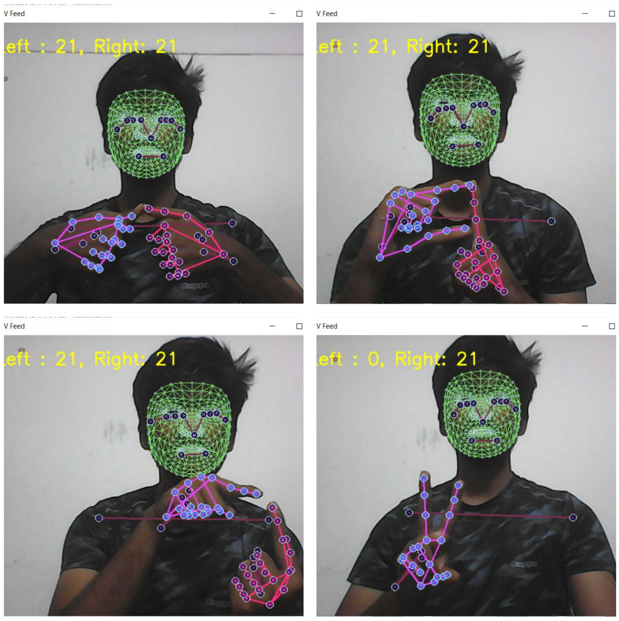


Fig. 1. Some of the Signs of ISL in the dataset

3.1 Architecture

The hand-code motions in sign language are well-organized, and each gesture/sign has a distinct meaning, thus the word is represented. Data acquisition, Keypoint Extraction and Classification or Detection of signals to equivalent text are the processes of the system. The following Figure 2 depicts the architecture of the proposed model. In this project, Mediapipe module is used for keypoint extraction. The dataset is created with these keypoints as NumPy array which is used for training. For the categorization of the extracted characteristic features, the LSTM model is utilised. A cell, an input gate, an output gate, and a forget gate for feedback make up an LSTM unit or neuron. These three gates regulate the flow of data into and out of the cell. The value of the cell is kept track of across arbitrary time periods. The input gate monitors the passage of the most recent extended value into the cell block. The forget gate controls how long a value is stored in the cell. The output block, on the other hand, is in charge of the value utilised to compute the LSTM unit's activation. The parameters of LSTM used are input_shape, activation, return_sequences. The LSTM Architecture is represented in Figure 3.

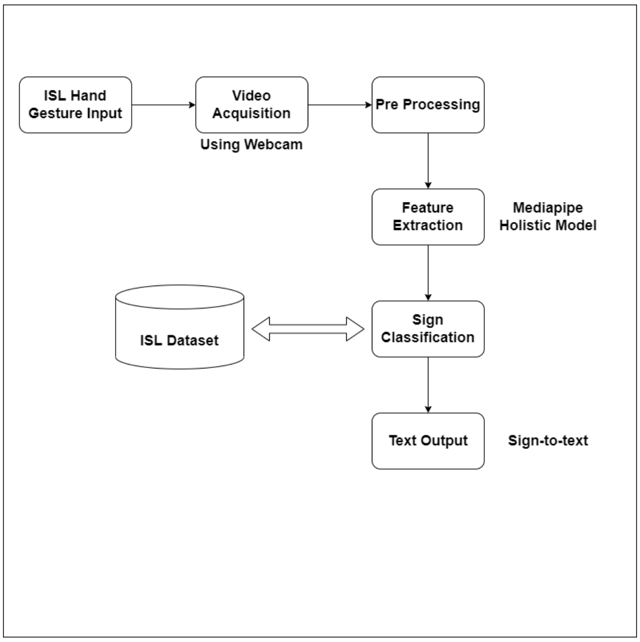


Fig. 2. Architecture of the Model

3.2 Algorithm

Below is the step by step algorithm for overall process for the proposed system;

```
BEGIN
    1. Dataset videos are created for alphabets and common words
in ISL.
    2. The recorded videos get converted to the image frames using
OpenCV.
    3. Training.
    (3.1) Keypoint Extraction done with Mediapipe.
    (3.2) LSTM classification(model, input_shape).
    4. Training and Testing.
    (4.1) Signs are detected as the equivalent text form.
END
```

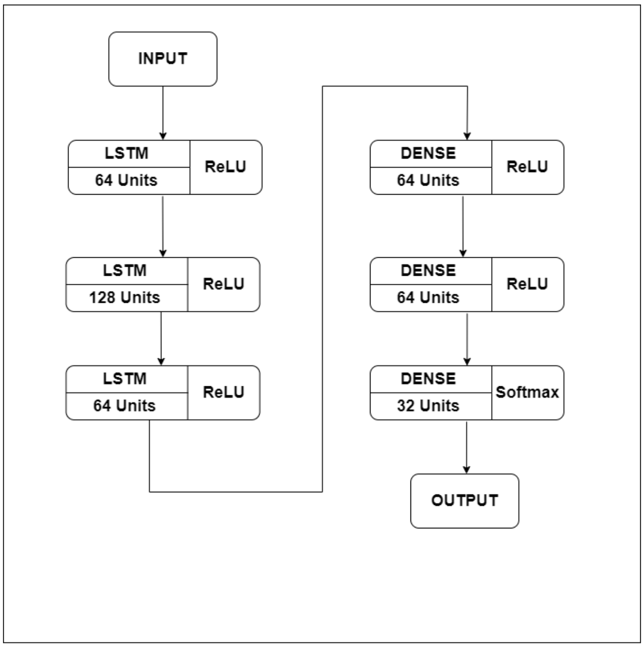


Fig. 3. LSTM Architecture

The following Figure 4 depicts a simplified flow diagram of the proposed system concept.

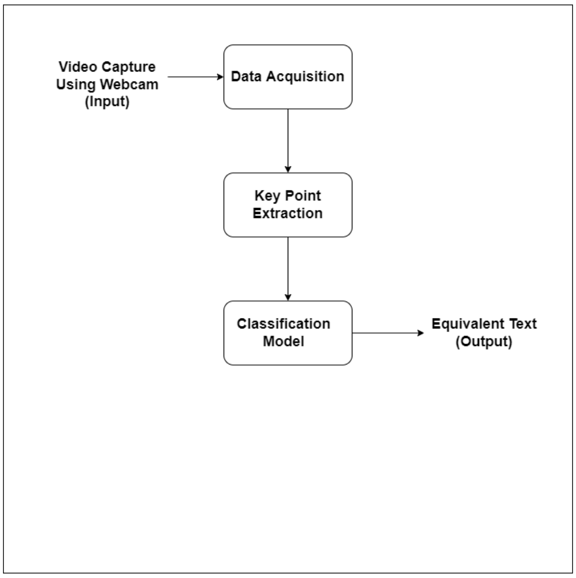


Fig. 4. Flow Diagram of the Model

4. METHODOLOGY

The methodologies of the various phases in the proposed system were discussed below.

4.1 Dataset Acquisition

This is done with OpenCV library. Hand gestures and signs were taken in video format and converted to a series of image frames. In this system, the self-customized dataset is created for ISL signs containing different lengthened gesture videos. For every alphabets and words present in the dataset, 30 videos has been recorded. Each video is parsed into 30 frames. Then, from these 30 frames, keypoint extraction is done. Then, these keypoints values which are in the form of NumPy array are stored in the data store, which acts as Dataset and used for training. The datset comprises 26 alphabets and 55 words signs in ISL.
As earlier mentioned, there is no standardized dataset for ISL. The dataset used here is Indian Sign Language Database, which is created with of resolution 1280 x 720 (0.92MP).

	Count
Alphabet	26
Words	55
Total	81

Table: 1 Dataset Overview

It contains 21,87,000 files: Python files contains NumPy array values. The Dataset contains gesture videos for 26 alphabets and 55 words. For each signs, 30 different videos were recorded with different background and lighting. In this way, the model can be trained more accurately and can be adapted to the real-time. Finally, Keypoints extracted are stored as NumPy Array Values.

	Videos	Frames - Keypoint Data Files
Alphabet	780	23,400
Words	1,650	49,500
Total	2430	72,900

Table: 2 Total Dataset Collection

4.2 Keypoint Extraction

Features are extracted from the video frames collected from the recorded videos exhibiting dynamic signs utilising the MediaPipe module in the suggested model. It makes use of a machine learning pipeline that consists of numerous models that operate together. A Hand Landmark Model that returns high-fidelity 3D hand keypoints from the cropped picture region determined by the palm detector. Similarly, MediaPipe Face uses a face detector algorithm with a Face Landmark Model. Thus, the extracted keypoints are stored NumPy array as python files and get stored. Figure 5 represents the Keypoint extraction process with Mediapipe module.

4.3 Classification Model

The built system that classifies the sign uses the keypoints extracted from the preceding process as an input. A classification tool is an LSTM network. The image is categorised as text with the help of the LSTM model. The advantage of utilising LSTM for categorization is that it eliminates the need for manual input feature engineering. The neural model network is taught to categorise hand motions during the training phase. Image frames from movies are employed for detection of indicators during the testing phase and in real time.

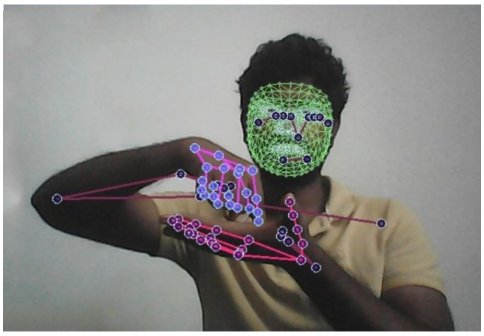
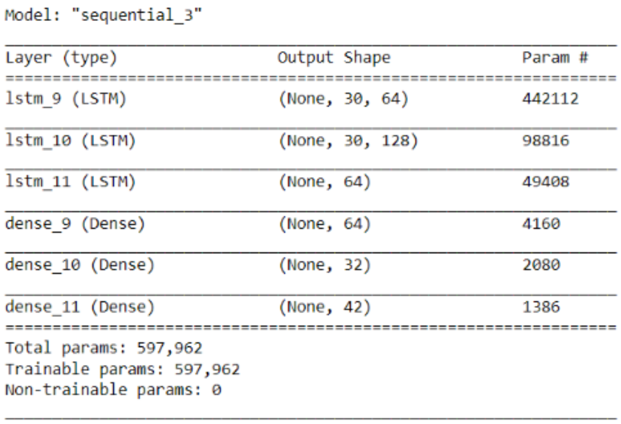


Fig. 5. Keypoint Extraction Process

```
model = Sequential()
model.add(LSTM(64,return_sequences=True,
    activation='relu', input_shape=(30,1662))
model.add(LSTM(128, return_sequences=True,
    activation='relu'))
model.add(LSTM(64, return_sequences=False,
    activation='relu'))
model.add(Dense(64, activation='relu'))
model.add(Dense(32, activation='relu'))
model.add(Dense(len(actions), activation='softmax'))
```



```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm_9 (LSTM)                (None, 30, 64)            442112
_________________________________________________________________
lstm_10 (LSTM)               (None, 30, 128)           98816
_________________________________________________________________
lstm_11 (LSTM)               (None, 64)                49408
_________________________________________________________________
dense_9 (Dense)              (None, 64)                4160
_________________________________________________________________
dense_10 (Dense)             (None, 32)                2080
_________________________________________________________________
dense_11 (Dense)             (None, 42)                1386
=================================================================
Total params: 597,962
Trainable params: 597,962
Non-trainable params: 0
_________________________________________________________________
```

Fig. 6. Classification Model Neural Network

5. TEST RESULT AND PERFORMANCE EVALUATION

In this project, the LSTM model is built with Mediapipe for keypoint extraction. The goal is to propose a way to help the community of the dead and deaf communicate. For this work, employed a self-recorded bespoke dataset due to the limited number of standardised datasets available for ISL. There are a total of 2430 videos in the dataset. The data set comprises of many indications, each of which was recorded multiple times. With a variable camera position, background, and lighting, each sign gesture action was unique. The lexicon of alphabets and other ordinary terms are included in the self-recorded collection of video signs. In this methodology, a new enhancement is to test the models using longer gesture films ranging of 30 frames per video with a resolution of 1280 x 720 pixels.

Dataset	Accurcay
Alphabet	93.8%
Words	82.3%
Entire Dataset	84.1%

Table: 3 Dataset Accuracy Overview

For the current scenario, accuracy of 84.1% is obtained with the created Dataset and the built model in the testing phase.

During testing phase, the Alphabet and the Words classification were done by the model. As a result, common words like me, understand, days of the week, I, etc were prominently classified as correct and with no confusion in the model. In the case signs like week, alphabet b, etc were classified incorrect and the model confused these words with certain other words.

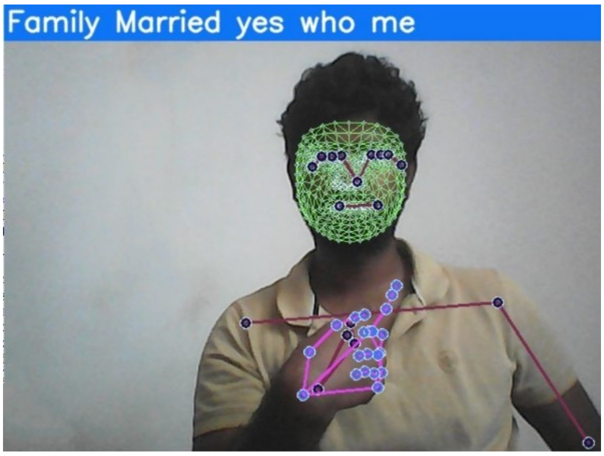


Fig. 7. Correct Detection of the word Me

Figures 7 and 8 depicts Correct and Wrong Detection of words.

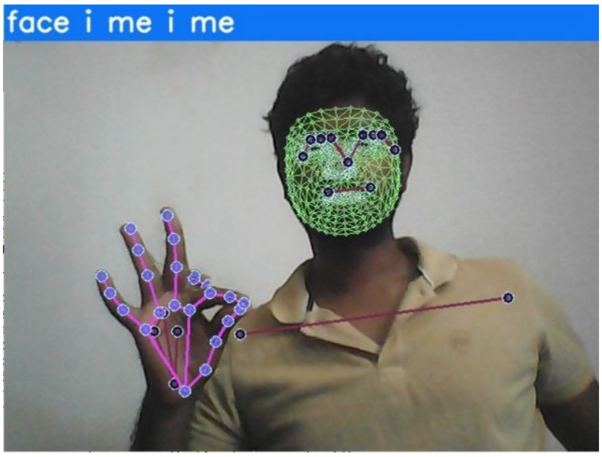


Fig. 8. Incorrect Detection of the word Week

The performance of the built detection system is analysed by its precision, recall, and F1- score.

Precision = T P/(T P + F P)
Recall = T P/(T P + F N)
F1 score = (2 Precision Recall)

The below table provides Performance insights on some Alphabets classification by the Model.

Signs	Precision	Recall	F1-Score	Support
a	1.0	1	1	7
b	0.89	1	1	8
c	1.0	1	1	9
d	1.0	1	1	5
e	0.91	1	1	13
f	0,9	1	1	8
g	1.0	0	1	8
h	1.0	1	1	8

Table: 4 Performance Metrics for some Alphabets in the dataset

The below table provides Performance insights on Words Classification by the Model.

Signs	Precision	Recall	F1-Score	Support
bye	1.0	1	1	9
bad	1.0	1	1	9
good	1.0	1	1	7
man	1.0	1	1	8
child	0.86	1	1	6
day	0.9	1	1	9
easy	0.88	1	1	8
food	0.9	1	1	9
hello	0.89	1	1	8

Table: 5 Performance Metrics for some Words in the dataset

The graphs present below depicts the classification of words and alphabets. The straight line present in the graph represents the correct matching of the words and the alphabets. The scattered plots present in the graph are the mismatched ones.

The Figure 9 shows Alphabet Tested graph and the Figure 10 shows Words Tested graph.

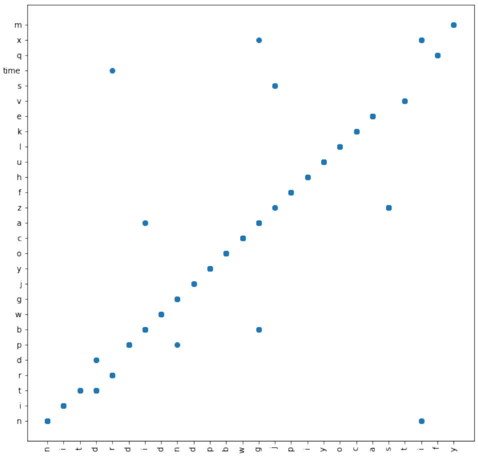


Fig. 9. Alphabet Classification Graph

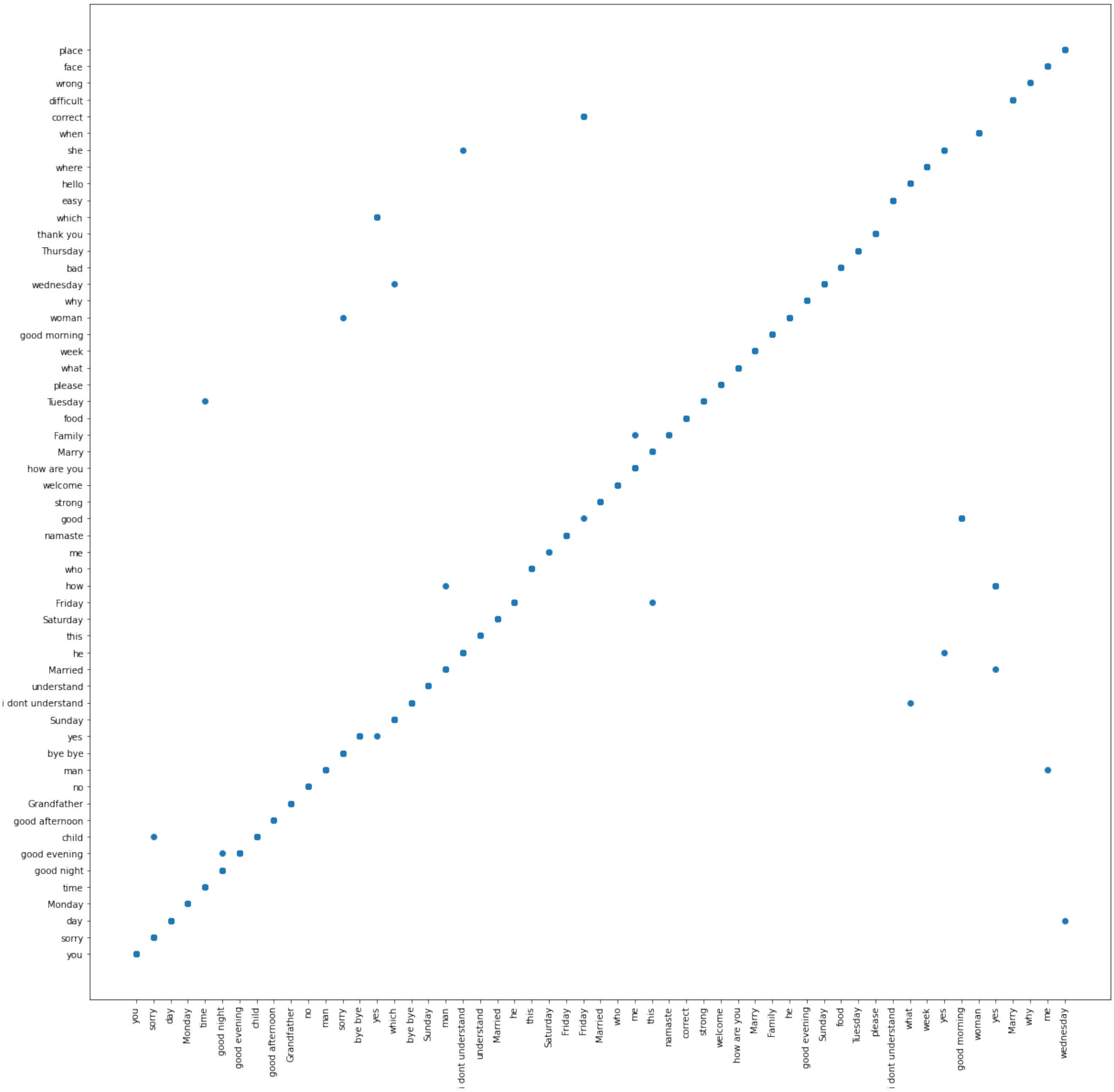


Fig. 10. Words Classification Graph

6. CONCLUSION

A primary advantage of this model is that it is designed to be an interface that functions in real time and would be available to masses. In the field of gesture recognition, recent breakthroughs in deep learning and technology have resulted in significant progress. The employment of the LSTM deep neural network model for sign identification improves hearing and speech impaired people's ability to interact with others. The major goal of this project was to create a system interface for ISL utilising self-created and customized dataset to break down communication barriers between the deaf-dumb people and the general public. The challenge may be absolutely computerized and prolonged for textual content to speech conversion. Also, exclusive fashions may be attempted for the reason of class and detection.

7. FUTURE SCOPE

The scope of this project is tremendous and there are multiple improvements that can be done and added to the project in the future. Dataset can be extended for other common words and the model can be utilised. The key point extraction can be further refined by maximizing the image frame size to 60. And increasing the Dataset size thus by increasing the video counts with multiple lighting and camera angle conditions can improve the accuracy and efficiency of the model.

The primary future scope of the project is that, the developed system can be designed and adapted to a specific environments like hospitals, banks, etc. Improving the performance of the device also can be done. This can also be developed into a mobile application, it would ensure that the model would be easily accessible and effectively distributed amongst the target audiences.

8. REFERENCES

[1] Bencherif M A. (2021) ‘Arabic Sign Language Recognition System Using 2D Hands and Body Skeleton Data’, IEEE Access, Vol. 9, pp. 59612-59627.

[2] Dhiman R., Joshi G. and Rama Krishna C. (2021) ‘A deep learning approach for Indian sign language gestures classification with different backgrounds’, Jouranal of Physics: Conference Series, ICMAI, pp. 1742-6596.

[3] Fernandes L, Dalvi P, Junnarkar and Bansode M. (2020) ‘Convolutional Neural Network based Bidirectional Sign Language Translation System’, Proc. Third International Conference on Smart Systems and Inventive Technology, pp. 769-775.

[4] Giulio Marin, Fabio Dominio and Pietro Zanuttigh. (2014) ‘Hand Gesture Recognition with Leap Motion and Kinect Devices’, Proc. IEEE International Conference on Image Processing (ICIP), pp. 1565-1569.

[5] Intwala N., Banerjee A., Meenakshi and Gala N. (2019) ‘Indian Sign Language converter using Convolutional Neural Networks’, Proc. IEEE 5th International Conference for Convergence in Technology (I2CT), pp. 1-5.

[6] Kar A. and Chatterjee P.S. (2015) ‘An Approach for Minimizing the Time Taken by Video Processing for Translating Sign Language to Simple Sentence in English’, Proc. International Conference on Computational Intelligence and Networks, pp. 172-177.

[7] Kim C.J. and Park H.M. (2021) ‘Per-frame Sign Language Gloss Recognition’, Proc. International Conference on Information and Communication Technology Convergence, pp. 1125-1127.

[8] Lee B.G. and Lee S.M. (2018) ‘Smart Wearable Hand Device for Sign Language Interpretation System With Sensors Fusion’, IEEE Sensors Journal, Vol. 18, No. 3, pp. 1224-1232.

[9] Liu T., Zhou W. and Li H. (2016) ‘Sign language recognition with long short-term memory’, Proc. IEEE International Conference on Image Processing (ICIP), pp. 2871-2875.

[10] Mathavan Suresh Anand, Nagarajan Mohan Kumar and Angappan Kumaresan. (2016) ‘An Efficient Framework for Indian Sign Language Recognition Using Wavelet Transform’, Circuits and Systems and Scientific Research Publishing, Vol. 07, pp. 1874-1883.

[11] Muthukumar K., Poorani and Gobinath S. (2018) ‘Extraction of Hand Gesture Features for Indian Sign languages using Combined DWT-DCT and Local Binary Pattern’, International Journal of Engineering Technology, Vol. 07, pp. 316-320.

[12] Pahuja D. and Jain S. (2020) ‘Recognition of Sign Language Symbols using Templates’, Proc. International Conference on Reliability, pp. 1157-1160.

[13] Pan W., Zhang X. and Ye Z. (2020) ‘Attention-Based Sign Language Recognition Network Utilizing Keyframe Sampling and Skeletal Features’, IEEE Access, Vol. 08, pp. 215592-215602.

[14] Shrinidhi Gindi, Amina Bhatkar, Hasan Haider S. and Ramsha Ansari. (2020) ‘Indian Sign Language Translator Using Residual Network and Computer Vision’, International Research Journal of Engineering and Technology, Vol. 07, No.07.

[15] Vipul Brahmankar, Nitesh Sharma, Saurabh Agrawal, Saleem Ansari, Priyanka Borse and Khalid Al Fatemi. (2021) ‘Indian Sign Language Recognition Using Canny Edge Detection’, International Journal of Advanced Trends in Computer Science and Engineering, Vol. 10, No. 03.