

Deep Learning Based Indian Sign Language Classification

Suvidhi Bam [1]

Class of 2022-23
Mayo College Girls' School
suvidhibam5@gmail.com

Reetu Jain [2]

Chief-Mentor and Founder,
On My Own Technology Private Limited Mumbai,
reetu.jain@onmyowntechnology.com

Abstract: *Special-abled people use sign language as their communication link. There are different variations of sign language available. Our paper focuses mainly on Indian Sign Language. Indian Sign Language uses two hands for showing letters. Also, the word spoken consists mainly of two hands. The data set of the letters and words available for the Indian Sign Language is of low quality. We have created data consisting of all letters and a few select words. Appropriate preprocessing is applied to the collected image data. The MediaPose Hands model is used to store the 3D coordinates of the hand. The coordinate data is used as an input to the DNN model. The DNN model is composed of 5 hidden layers with a single dropout layer to avoid overfitting the data. The model has an accuracy of 96% which is better compared to the usual CNN model available for sign language classification. The model can complete a single prediction in 34ms. The Mediapose hands model can complete the landmark extraction in 60ms. On average the proposed model can process a single frame in 100ms giving us 10 frames per second. Which makes the model usable in real-time applications as well.*

Keywords: Sign Language, Machine learning, MediaPipe, ISL, ASL, Hands, DNN, Neural Network

1. Introduction

People with hearing and speech impairments utilize sign language, a kind of visual gesture and sign language, as a form of communication. The majority of communities that serve the deaf and hard of hearing rely on sign language interpreters to communicate with others. These individuals thus experience social isolation. Professional interpreters are their only means of communication, yet these services are typically pricey and not at all accessible. These interpreters are also infrequently accessible. It is therefore incredibly desirable to have an automatic recognition system. The substantial number of persons who use sign language as their primary form of communication can benefit significantly from automation in the domain of sign language recognition (SLR). Successful implementation of such a system would make it possible for specially-abled people to communicate flawlessly. The idea is to come up with a standalone device that can interpret the sign language with no extra time/attachments required. This requires the device to be lightweight and compact. This requires the model to be less resource hungry and hence we came up with a custom DNN model that relies on the hand landmarks instead of the traditional CNN model which is way more resource hungry. Another drawback of the CNN-based model is that the time required to process each frame is very big. Thus making the solution not good for real-time applications. The MediaPose Hands model is used to extract the hand landmark data from the collected data set. The MediaPose library is optimized so the landmark extraction process is very fast. This data is used to train the DNN model. Once trained the DNN model will rely on MediaPose ML models. The image data collected is first sent to the Mediapose Hands model. The result is then sent to the trained DNN model. The model outputs labels that represent letters or words based on the input coordinates.

2. Literature Review

Most research papers based on Machine learning suffer from a lack of data availability. But Sign Language is a domain for which data is available freely. Custom data can be generated at will by clicking images of a particular sign. This enables the generation of data in almost all known conditions in which the model is expected to perform thus improving the accuracy of the model. C. Chuan et al.[1] proposes a portable, inexpensive 3D motion sensor-based American Sign Language identification system. Compared to the Cyberglove or Microsoft Kinect utilized in earlier experiments, the palm-sized Leap Motion sensor offers a considerably more convenient and cost-effective option. The 26 letters of the English alphabet in American Sign Language are classified using the derived features from the sensory input using k-nearest neighbors and support vector machines. Imran Hussain et al.[2] has used image processing-based techniques to create a real-time gesture recognition system that can utilize the video camera linked to the computer. This enables specially-abled people to communicate with the computer. PCA(Principal Components Analysis) is mainly used to differentiate the same images precisely. Neha V. Tarvari et al.[3] has proposed a method for recognizing Indian Sign Language using hand movements. An innovative, natural, and user-friendly method of interacting with the computer that is more accustomed to people is provided by hand gesture recognition systems. The suggested method can recognize the signer's photos that are dynamically taken during testing. We have used a basic web camera to perform this strategy and record hand gesture photographs. Different signs are recognized by artificial neural networks, which then translate them into text and audio formats. B. Divya et al.[4] says that Electromyography (EMG) detects the electric potential generated by muscle cells when these cells are electrically or neurologically activated. EMG data from the subjects who are performing ISL were collected using BIOPAC MP-45. From the collected EMG signals features like mean absolute value (MAV), simple square integral (SSI), standard deviation (STD), root mean square (RMS), average amplitude change (AAC), maximum (MAX), minimum (MIN), average power in the channel (P) are extracted using MATLAB algorithm. A support vector machine classifier is used to classify the different gestures. The study shows that a real-time ISL recognition system using EMG has an accuracy of 90%. Muhammad Al-Qurishi et al.[5] has done a comprehensive overview of automated sign language recognition based on machine/deep learning methods and techniques published between 2014 and 2021 and concluded that the current methods require conceptual classification to interpret all available data correctly. Sakshi Sharma et al.[6] has come up with an SLRS(Sign language recognition system) which bridges the gap between the signer and the non-signer Indian communities. They have used data augmentation to increase the quality of the data. A CNN-based model was used for feature extraction and classification of ISL gestures. They achieved an accuracy of 92.43%. Deep Kothadiya et al.[7] Proposes a deep learning-based model (using deep learning models LSTM and GRU) that detects and recognizes the words from a person's gestures. It involves taking a video of the person making hand gestures, processing it, and passing it to the proposed model, which predicts words one by one. The system then generates a meaningful sentence out of those words that can then be converted into the language selected by the communicator. The proposed model achieved an accuracy of 97%.

3. Proposed Methodology

Data acquisition

There is an ample amount of sign language datasets available online. But the quality of the dataset available is not good in terms of variability. An ideal dataset should have images taken at different angles and distances. This is not the case with most of the datasets available online. Most of these data sets are images that are segregated into individual letters and words for which the model is to be trained. There is really low clarity regarding the Indian sign language since there exist many variants amongst the community. So instead of relying on the existing dataset, we decide to create our own data set. A simple OpenCV-based application was created to capture the images continuously. So a data set consisting of 500 images was created for each letter and word. During data collection, it was made sure that the hand was moving constantly so that all the possible angles and distances were covered in the dataset; this ensures that the trained model is unbiased on the distance and the orientation of the

hand. The lighting conditions were also changed during the image capture process; this is necessary as the MediaPose library which was used to extract the hand landmarks is a bit sensitive to lighting conditions.

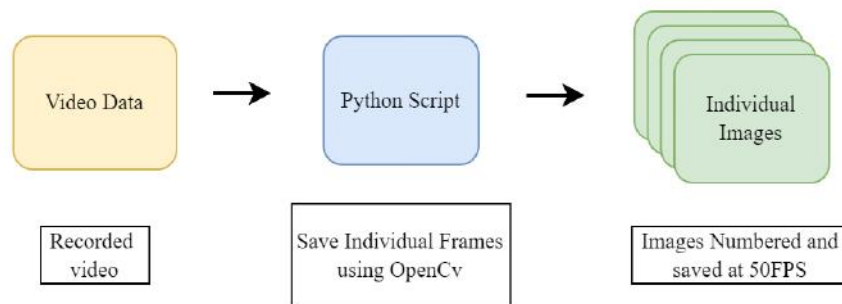


Fig 1. Steps used in Data Acquisition

Data Pre-processing

We can directly use the image to train a CNN-based model which can then be used to classify the letters. This approach will be really inefficient when we have pre-trained models like MediaPipe which has been trained on huge data sets. The MediaPose library is very reliable so we in turn rely on the MediaPose Hand's model to extract the hand landmarks. The CNN model also turns out to be very slow as the input letter will have the entire image pixel data. MediaPose Hands model uses a palm detection model in the background to detect the palm first as it is easy to detect objects with rigid shapes instead of trying to detect fingers as the fingers can have articulated shapes. The Hand model detects 21 landmarks for each hand. During preprocessing we feed the image to a python script that uses MediaPose's hands model to detect the landmarks of the hands present in the image. An image present in the data set is only considered for landmark detection if two hands are detected in the image. This restriction is imposed as all the letters present in the ISL are signed using 2 hands. Landmarks of both hands are first calculated and then the 3D world landmarks are detected.

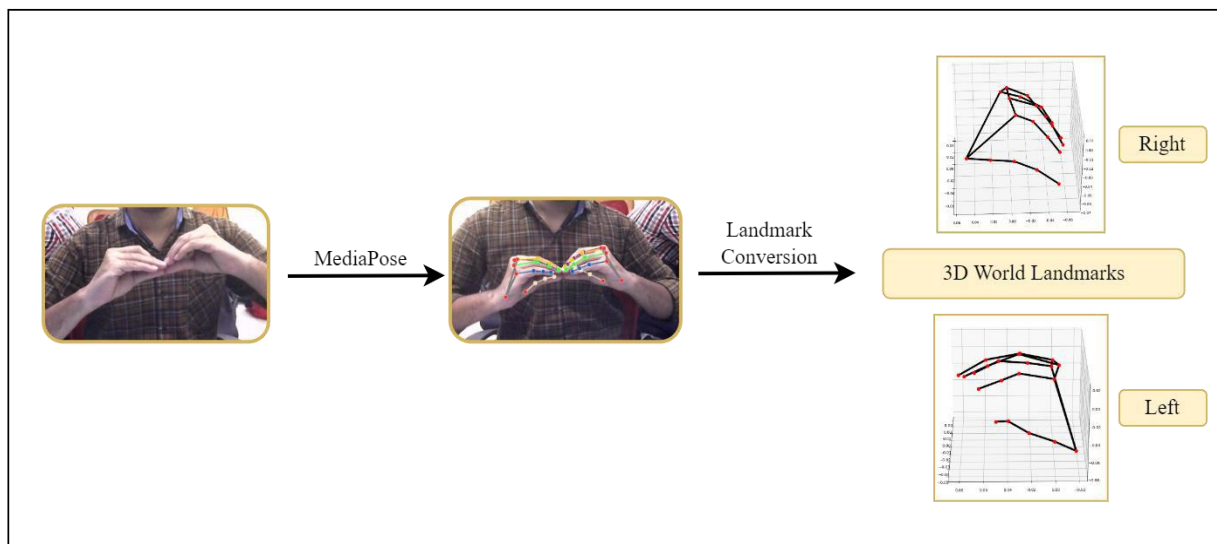


Fig 2. Pre-processing used to get the 3D world landmark data

Since the 3D world landmark is calculated with the bottom of the wrist at the origin we don't need to normalize the location of the hand in the frame. Ideally, we would need to normalize the size of the hand as different people can have different hand sizes also the hand can be at different distances from the camera. But we are not compensating for these changes as during the dataset creation we made sure that the images clicked were in all

orientations and distances from the camera. The 3D world coordinates for both hands are converted to a Numpy array. The separate arrays for the left and the right hand are concatenated to form a single array. This single array is then reshaped to create a 1-dimensional array. This step is repeated for all images of all letters in the dataset.

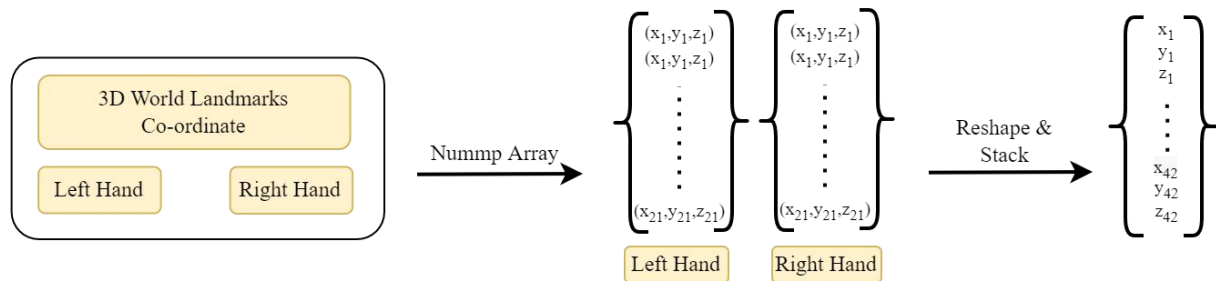


Fig 3. Reshaping 3D landmark data from the images

The dataset for every letter is composed of 500 images. So all the 500 images are made to pass through the above-mentioned process and all the resultant 1D arrays are stacked vertically. This NumPy array is saved to a .npy which can later be accessed for training the model.

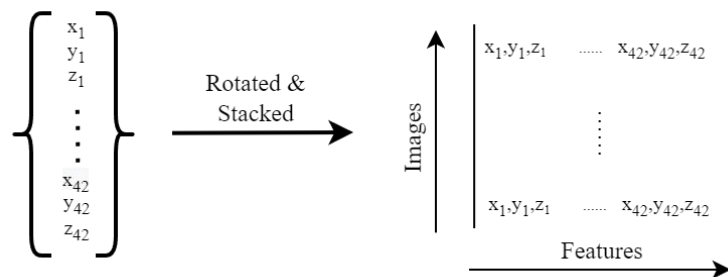


Fig 4. Shape of .npy array save for each letter

The NumPy arrays created for each letter had the shape $(500,42)$. Where 42 was the number of features that were going to be used for the model training. 21 of these input features were for the left hand and the other 21 were for the right hand. Converting the image data to a Numpy array compresses the dataset and makes it easy to share the data set. We can also increase the size of the dataset by just vertically stacking more such NumPy arrays which are created using images of a particular sign.

4. Machine learning model

There were 2 main choices when training the ML model for sign language detection. The first one is KNN(K-Nearest Neighbour) and the second one is Neural network(NN). Since the output label is made up of 26 minimum separate classes the KNN model was not selected as KNN models work best for classification problems with less number of output labels. We finalized using a DNN model for training the data. Provided the complexity of the input variables a simple Neural Network was not going to be optimal. So we opted for a Deep Neural network composed of 5 layers. The first input layer had a shape of 42 which corresponds to the number of input features for 2 hands generated by the MediaPose Hands model. The input layer was followed by 2 fully connected Dense layers and a Dropout layer with a 50% dropout and finally the output layer which was made up of 26 letters. The Numpy arrays of all the letters and the words selected were of the same shape $(500,42)$. This is to ensure that no biasing is induced due to the length of the arrays available for each letter.

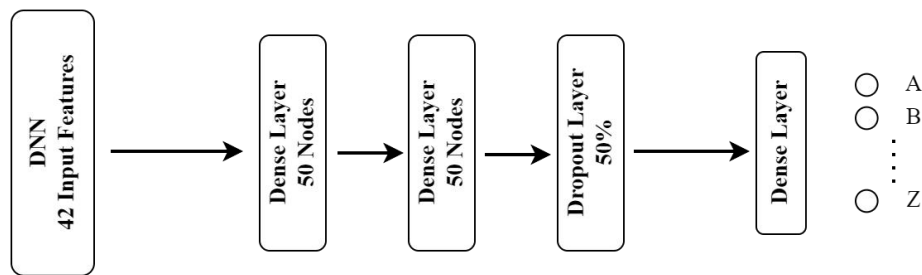


Fig 5. Sequential model created using Keras API's

The total size of the array for all the letters combined was 13000. The output label array was also generated using one-hot encoding for every letter. This array also has the same shape. 10% of the total array was withheld for testing the model. The Keras API is used for training the model. The Keras API by default shuffles the training data so any bias related to the sequence of training was not present.

The training dataset was used to train the Sequential model after it had been created. During training, Adam's optimizer was applied. The training data set was jumbled before the model was trained. The sklearn train test split function was used to shuffle the data. Since the DNN model needs data in terms of 0s and 1s to function properly, input data containing the labels of the letters were also converted to categorial. Once the model was trained all the performance metrics related to a standard DNN model were calculated and are presented in the following section. Once the model is trained we can use the model with a live video feed where every frame from the video will have to be passed through the MediaPipe Hand model and the 3D landmark generated will be the input to our mode. Thus allowing for a real-time sign recognition system.

5. Conclusion

The trained model had an accuracy of 96%. Since the model also relies on the MediaPose hands model, The accuracy during real operation is also dependent on the accuracy of the MediaPose hands model. During testing, it was observed that occasionally the MediaPose will detect no hands even though hands were in front of the camera. This was mostly because of the lighting conditions. Although this does not affect the accuracy of the model it hinders its reliability. The MediaPose hands model has an accuracy of 95.7% the overall accuracy of the system will be the multiplication of both which turns out to be 91.7%. Since a few letters like 'C' use just a single hand(right) during the training process the left-hand coordinates were set to be 0. This causes the model to perform a bit inaccurately.

Future Scope

The current model utilizes images as input to determine if a hand is present in the model. If yes the model outputs a letter or a word based on the confidence score. But for most sign language-related communication, the signer uses gestures and facial expressions instead of using letters. To capture the facial expressions we can utilize the MediaPose Face library. This does not fix the issue where the model cannot recognize gestures that exist for more than a single frame. To fix this problem we can train an LSTM-based model which can learn the hand landmarks from different frames and convert them into words. This would not affect the working of the existing single frame-based model. In addition to this, we can also run an auto-correction feature in the background which can enhance usability by correcting any wrong words or letters that were caught by the model. Currently, the model is trained directly on the coordinate of every landmark. This approach is not the most optimal. We could instead apply

feature engineering to convert the landmark coordinates into more meaningful distance metrics thus reducing the number of input variables and thus probably increasing the accuracy as well as increasing the speed of the model.

Acknowledgment

I would like to express my gratitude to the mentors of On My Own Technology Pvt. Ltd. for extending their help in carrying out this project.

References

1. C. Chuan, Eric Regina, and Caroline Guardino. American Sign Language Recognition Using Leap Motion Sensor. *13th International Conference on Machine Learning and Applications*. 2014.
2. Imran Hussain, A. K. Talukdar, Kandarpa, and K. Sarma. "Hand Gesture Recognition System with Real-Time Palm Tracking Using American Sign Language. *IJCRT Volume 6, Issue 2* April 2018.
3. Neha V. Tavaris, Prof. A. V. Deorankar, and D. Chatur. Hand Gesture Recognition of Indian Sign Language to aid Physically impaired People. *International Journal of Engineering Research and Applications*. April 2014.
4. B. Divya, J. Delpha, and S. Badrinath. Public speaking words (Indian sign language) recognition using EMG. *International Conference On Smart Technologies For Smart Nation (SmartTechCon)*. 2014.
5. Muhammad AL-Qurishi, Thariq Khalid, and Riad Souissi. Deep Learning for Sign Language Recognition: Current Techniques, Benchmarks, and Open Issues. *IEEE Access (Volume: 9)*.
6. Sakshi Sharma and Sukhwinder Singh. Recognition of Indian Sign Language (ISL) Using Deep Learning Model. *Wireless Personal Communications volume 123*, pages671–692 (2022).
7. Deep Kothadiya, Chintan Bhatt, Krenil Sapariya, Kevin Patel, Ana-Belén, Gil-González and Juan M., Corchado. Deepsign: Sign Language Detection and Recognition Using Deep Learning. *Recent Advances in Computer Science & Engineering*. June 2022.