ASL Sign Language Alphabet Recognition using Ensemble Convolution Neural Networks

Chetan Krishna Palicherla
Aditya Varma
Abhinav Baddireddy
Under the Supervision of Prof. Santhi K.

***Abstract* - Sign Language is the most popular tool for people with hearing impairment to communicate with anyone. To solve this problem many researchers used multiple approaches, including Convolutional Neural Networks (CNN), which gave high accuracies. This paper aims to build an ensemble CNN model, by using 4 different CNN models. The proposed model performed not very well with an accuracy of 79.68%, average precision of 80.2%, and an average recall of 77.7%.**

I.INTRODUCTION

According to the World Health Organization (WHO), as of 1st April 2021, there are about 466 million people (432 million adults and 34 million children) with hearing disabilities around the globe, which is approximately 5% of the world population. WHO predicts that by 2050 the number would increase to 700 million people, which is 10% of the world's population. Due to unsafe hearing practices, more than 1 billion adults are at risk of permanent hearing loss. In this scenario, having an efficient communication tool for people with hearing disabilities, and sign language is the most used tool in this context. Sign Language is a language that uses hand and facial gestures to communicate with one another. There are different kinds of sign language around the world, like American Sign Language (ASL), British, Australian, and New Zealand Sign Language (BANZSL), Chinese Sign Language (CSL), French Sign Language (FSL), Japanese Sign Language (JSL), Arabic Sign Language, Spanish Sign Language (LSE), Mexican Sign Language (LSM), Indian Sign Language (ISL). This paper will be focussing on American Sign Language (ASL).American Sign Language (ASL) is a Sign Language (SL) that is a predominantly used SL by the people of the United States of America (USA) and Canada. Apart from North America, many other countries like much of West Africa and parts of Southeast Asia also use predominantly use ASL for communication between people with hearing disabilities. It originated in the early 19th century in the American School for Deaf (ASD) in West Hartford, Connecticut, from a situation of language contact. ASL consists of several moments of the face, the torso, and the hands. ASL alphabets are a set of static and dynamic hand signs that represent the alphabet in the English Language.

Many researchers have used multiple approaches to building an efficient model for Sign Language Recognition. They have used Deep Learning, Convolutional Neural Networks, deepCNN, Image Processing feature extraction methods like Ostu Thresholding and Scale Invariance Feature Transform (SIFT), Principal Component Analysis (PAC), data augmentation, etc. And most of them achieved a good accuracy of greater than 90%. This paper proposes an ensemble CNN model, that used four different CNN models and combines them with using the hard voting method to predict the output. Ensemble Learning methods are known to increase the accuracy of most of the models.

II. LITERATURE REVIEW

1. A systematic review on hand gesture recognition techniques, challenges and application

This report summarised some of the research on HGR radar applications. Currently, the researchers rely mainly on the readily accessible radars produced by tech firms like Texas Instrument, Infenion, and Novelda. The algorithms for gesture detection and recognition have received a lot of interest since these systems are on chips. The focus has shifted in recent years from HGR algorithms based on signal processing to those based on deep learning. Variations of CNN in particular have demonstrated promising usefulness. Even though radar sensors have several benefits over other HGR sensors (such as cameras and wearable sensors), radar-based HGR is still not as widely used as its rivals. Real-time recognition algorithms and the development of tiny hardware require attention.

2. Hand Gesture Recognition with Depth Images: A Review

Among the 37 papers reviewed, 13 methods were used for hand localization, and 11 more were used for gesture classification.24 of the papers included real-world applications to test a gesture recognition system and a total of 8 categories of applications were used. Though five different types of depth sensors were used, the Kinect was by far the most popular (used by 21 of the papers). The Kinect also has available hand-tracking software libraries that were used by 8 of the papers, and the papers that used the Kinect tended to focus more on applications than on localization and classification techniques. Returning to the three questions posed in the Introduction: What methods are being used to achieve hand localization and gesture recognition with depth cameras? A total of 10 methods are commonly used for hand tracking and gesture recognition in the papers reviewed (2 for segmentation, 3 for tracking, and 5 for classification). The significance of the methods used is that while standard machine learning algorithms are used for gesture classification, hand localization – in particular hand segmentation – has more specialized approaches. Also, custom hand detection and tracking methods are being replaced by off-the-shelf solutions such as PrimeSense's NITE module for the OpenNI framework.

3. A Survey of Deep Learning-Based Human Activity Recognition in Radar

As an active system for human activity recognition, radar has many unique advantages. Few models have different characteristics for identifying human activities and there is a trend of combining multiple models to better learn the features of human activities. Doppler radar can obtain Doppler information for HAR while FMCW radar provides both range and Doppler information.UWB radar has a high-range resolution and is capable of distinguishing the scattering centers of the human body. Furthermore, by classifying radar echoes into three different forms: 1D, 2D, and 3D, we discuss the development of deep learning-based HAR in radar. Various deep learning techniques designed especially for 1D/2D/3D radar echoes have been discussed, and the experiment results demonstrate the feasibility of such techniques.2D radar echoes, especially time–Doppler maps, are more commonly used for radar-based HAR because they are more intuitive, and contain sufficient activity information. Though the adoption of radar for HAR is still lagging behind vision-based technologies, we should be optimistic about the potential of radar-based HAR techniques because of radar's unique advantages such as environment insensitivity and better privacy protection.

4. A Deep Convolutional Neural Network Approach for Static Hand Gesture Recognition

This study makes use of the deep learning tool's capacity to recognize hand positions from raw RGB photos. The computational overhead associated with hand posture identification using conventional methods is reduced by the proposed CNN architecture since it does not need the detection and segmentation of hands from the collected pictures. Additionally, even with very slight interclass changes, the model can automatically infer the probable attributes that distinguish the hand postures. Through five-fold cross-validation, the effectiveness of the suggested strategy has been assessed on two publically accessible datasets. The performance study of the proposed CNN model utilizing statistical criteria like accuracy, precision, recall, and F1- score demonstrates its improved recognition capacity.

5. 3D-CNN based Dynamic Gesture Recognition for Indian Sign Language Modeling

In this paper, the authors sought to create a deep neural network that could represent and recognize common Indian hand gestures a sign language. The information was gathered from a variety of age and background groups. The modeling exercise for these dynamic motions is analyzed using the fundamental 3DCNN architecture. Experimental results validate the model's output concerning the acquired accuracy levels. Not everyone is familiar with hand. gestures in daily life. An issue is that every nation has its unique collection of symbols. global standardization Lack of level makes communicating outside of the nation difficult. Despite this, we've modeled the gestures in the hope of possessing a sizable vocabulary that could translate one nation's hand.

6. Hand Gesture Recognition Using PCA

In this paper, the authors built a Hand Gesture Recognition System based on a bare hand gesture system by using a database-driven system based on a skin color model and thresholding approach, They also used an effective template matching suitable for human robotics applications. The hand region was first separated with the help of the skin color model in the YCbCr color space, then the Ostu thresholding was applied to separate the foreground and background. Finally, Principal Component Analysis was used to build the model. The model performed with 100% accuracy, precision, and recall with controlled data and 91.43% accuracy, 94.12% precision, and 96.79% recall with images with less brightness than the original data.

7. Hand Gesture Recognition System Using Image Processing

This paper aims to build an efficient Hand Gesture Recognition System using the Scale Invariance Feature Transform (SIFT) algorithm, which has a high processing speed. It uses SIFT features which reduce the feature dimension vector to find the edges. This nullifies any effects of scaling, rotation, and any additional noise. SIFT and Principal Component Analysis (PCA) are then used to extract the key points of the hand gesture. This paper also explains the working of SIFT algorithm and PCA algorithm. It also discusses the Dimensionality Reduction technique and Classification methods. It finally presents the applications, limitations, and advantages of these systems.

8. Automated Indian sign language recognition system by fusing deep and handcrafted feature

This paper proposes an automated Sign Language Recognition System (SLRS), that overcomes the problem faced with identical hand orientation and multiple viewing angles. The model proposed by this paper is called Automated Indian Sign Language Recognition System for Emergency Words (AISLRSEW). It focuses on recognizing Indian Sign Language (ISL) which is often in emergencies. AISLRSEW uses features extracted from a Convolution Neural Network (CNN), local handcrafted features, and SIFT algorithm. This paper used a dataset with RGB clips of eight ISL emergency words – Accident, Call, Doctor, Help, Heat, Lose, Pain and Thief. The data were collected from 26 adults. The model with only CNN feature extraction got an accuracy of 100% on the training dataset and 90.29% on the test data, whereas combined with handcrafted features, the model got an accuracy of 100% on the training dataset and 94.42% on the test data.

9. Hypertuned deep convolutional neural network for sign language recognition

This paper proposed a DeepCNN architecture to recognize American Sign Language (ASL) alphabets. The sign language MNIST dataset was used to create the model. The data was augmented using scaling and rotations. The data architecture proposed by the paper was 3 convolutions followed by max pooling, and then a flattened layer and 3 dense layers. An accuracy of 99% for achieved by the model proposed by this paper.

10. Gesture Recognition Using Surface Electromyography and Deep Learning for Prostheses Hand: State-of-the-Art, Challenges, and Future

This review paper briefly introduces the advances of DL-based sEMG pattern recognition techniques for the prosthetic hand in recent years. Through the literature survey of DL application in sEMG recognition, some of the core techniques are highlighted, some of the most common challenges to be solved are analyzed, and some of the most possible development prospects are discussed. It could be found that gesture recognition techniques based on DL have great potential in using sEMG signals to accurately interpret an amputee's motion intention, which is of great significance to the development of the intelligent prosthetic hand. However, their difference, real-time usability, and long-term stability are still highly limited by many complex factors in these approaches. The high variability of sEMG, the lack of existing data, the limitation of hardware resources, and the lack of clinical evaluation conditions seriously affect the progress of pattern recognition techniques based on DL. In addition, the natural movements of the upper limbs are independent, continuous, and proportional activations of multiple DOFs, while the existing techniques can only use a limited number of patterns for discrete classification. These should be well-improved in the future real-time application of prosthetic hands.

11. Review of constraints on vision-based gesture recognition for human–computer interaction

Vision-based gesture recognition typically depends on three stages: gesture detection and pre-processing; gesture representation and feature extraction; and recognition. Vision-based gesture recognition typically depends on three stages. Detection includes various kinds of imaging systems, whereas pre-processing includes segmentation of gesturing body parts and occlusion handling. Gesture modeling is followed by the selection and extraction of appropriate features. Extracted features should be RST invariant, object view independent, and computationally inexpensive. A VBI is expected to enable a user to interact with a machine with natural human-to-human interactions in an unconstrained environment. This paper surveyed the major constraints in the different stages of a VGR system

12. Deep signature-based isolated and large scale continuous gesture recognition approach

In this article, they have proposed an effective temporal segmentation. They use optical-flow estimation using SpyNet and fed it to Gesture Signature descriptors. Spatio-temporal features are then evaluated with an SVM classification system. Experiment results prove that their Deep Signature model is robust and its performance exceeds state-of-the-art gesture characterization methods. well-known isolated action KTH data set and Chalearn ConGD.

13. An improved hand gesture recognition system using key points and hand bounding boxes

This manuscript presents four classification architectures for static hand gesture recognition with data processed by top-down pose estimation using HRNet.The experiments were conducted on three datasets called HANDS, SHAPE, and OUHANDS. Among four architectures, the two-pipeline architecture can learn the features combined with both hand-bounding boxes and key points. In our future work, it is better to use our methods with the data processed by pose estimation (HigherHRNet) to improve the results further. lightweight models to use this system in real-time applications.

14. Real-Time Hand Gesture Recognition Based on Deep Learning YOLOv3 Model

In this paper, they have proposed a lightweight model based on the YOLOv3 and DarkNet-53 deep learning models for hand gesture recognition. The developed hand gesture recognition system detects both real-time objects and gestures from video frames with an accuracy of 97.68%. Despite the accuracy obtained there is still room for improvement in the following model, as right now the model proposed detects static gestures. The proposed method can be used for improving assisted living systems, which are used for human–computer interaction both by healthy and impaired people. Additionally, we compared the performance and execution of the YOLOv3 model with different methods and our proposed method achieved better results by extracting features from the hand and recognizing hand gestures with accuracy, precision, recall, and F-1 score of 97.68, 94.88, 98.66, and 96.70%, respectively.

15. Dynamic Hand Gesture Recognition: A Literature Review

A thorough review of hand gesture recognition methods has been given by this survey. Several computer applications can benefit from hand gestures as an intriguing interactional insight. When employing them, two key questions must be addressed: What technology should be used to obtain raw data from the hand comes first. There are typically two technologies available for gathering this raw data. The first method is a glove input device, and the second method uses computer vision to get raw data. One or more cameras positioned in the environment record hand movement in a vision-based system. Which option to pick is a challenging decision because both sorts of solutions have benefits and drawbacks. It seems that vision-based systems are the greatest option for gathering raw data if they can get over some of their challenges and drawbacks. What recognition method will enhance accuracy and robustness while employing hand gestures is the second question that has to be addressed. There are many different recognition methods, however in particular situations when hand gesture recognition is needed, the options are limited since some algorithms work better for gesture recognition exclusively. These methods have been grouped into three major groups in this review: model learning algorithms, statistics, and feature extraction. Future research in hand gesture recognition has a lot of intriguing directions.

16. A brief review of vision based hand gesture recognition

This study presents an overview of vision-based hand gesture identification techniques. The area of vision-based hand gesture detection has made notable development in recent years. The ultimate objective of human-machine interaction on their terms requires more study in the fields of feature extraction, classification techniques, and gesture representation. Hand gesture recognition is a technology with a bright future. The surrounding gadgets will most likely support hand gesture interfaces sooner than one may anticipate

III. DATASET

Sign Language MNIST Dataset is used for conducting this experiment. Sign Language MNIST is an ASL alphabets dataset, consisting of sign images and labels. Alphabets A-Z are represented by 0-25. The dataset excludes J (9) and Z (25) as they include movement of the hand and can’t be captured in an image. This dataset has 27455 rows.

Each Image is of size 28x28 and is grayscale. The dataset is already cleaned and is on a clear background. So there is no need of separating the foreground and background before building the model, and there is no need of data cleaning also.

There is also little data imbalance in the output class, we can see the distribution of the labels in FIG. 1 and FIG, 2.

FIG. 1 BAR CHART OF LABELS

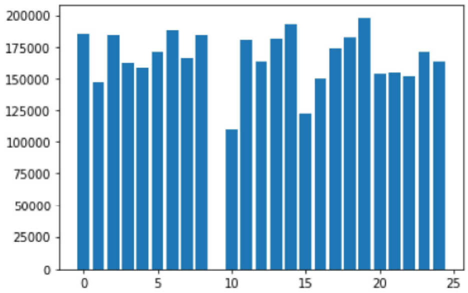
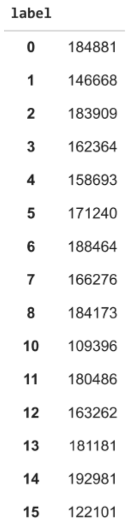


FIG. 2 DISTRIBUTION OF LABELS



label	
0	184881
1	146668
2	183909
3	162364
4	158693
5	171240
6	188464
7	166276
8	184173
10	109396
11	180486
12	163262
13	181181
14	192981
15	122101

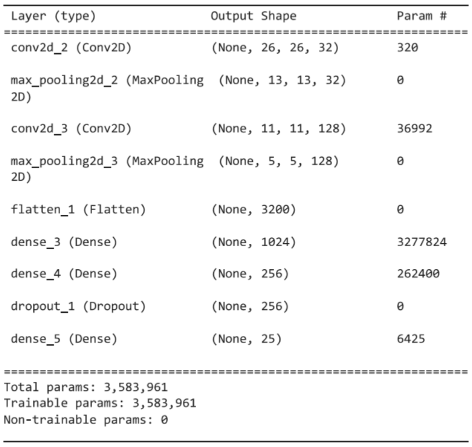
It can be observed from FIG. 1, that there are relatively fewer data points with output labels 10 (K) and 15 (P). But this is not a noticeable difference, hence was ignored.

IV. METHODOLOGY

The data in the dataset is set is presented as a label followed by 784-pixel values (28x28 pixels flattened). First, the dataset is converted into (27455, 28, 28) shape and then, each image is normalized by dividing each pixel by 255. The dataset was then divided into Test and Train Set.

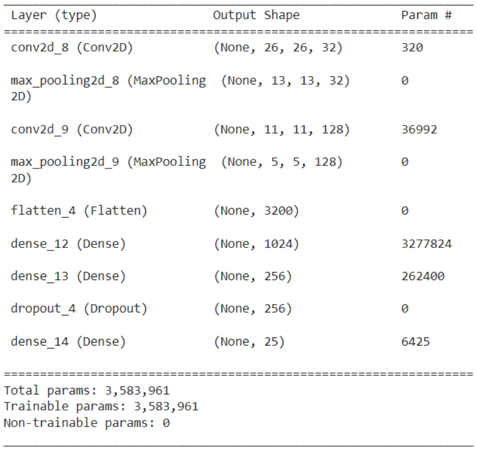
Then 7 different CNN Models were used for experiments, and four of them were selected with the best performance. The Architecture for these four CNN models is in FIG. 3, FIG. 4, FIG. 5, and FIG. 6.

FIG. 3 ARCHITECTURE OF MODEL 1



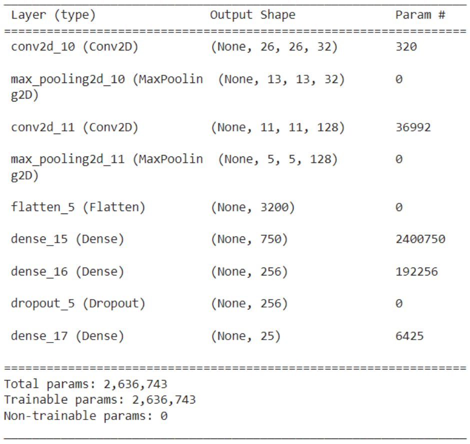
```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_2 (Conv2D)            (None, 26, 26, 32)        320

max_pooling2d_2 (MaxPooling  (None, 13, 13, 32)        0
2D)

conv2d_3 (Conv2D)            (None, 11, 11, 128)       36992

max_pooling2d_3 (MaxPooling  (None, 5, 5, 128)         0
2D)

flatten_1 (Flatten)          (None, 3200)              0

dense_3 (Dense)              (None, 1024)              3277824

dense_4 (Dense)              (None, 256)               262400

dropout_1 (Dropout)          (None, 256)               0

dense_5 (Dense)              (None, 25)                6425

=================================================================
Total params: 3,583,961
Trainable params: 3,583,961
Non-trainable params: 0
```

FIG 4. ARCHITECTURE OF MODEL 2



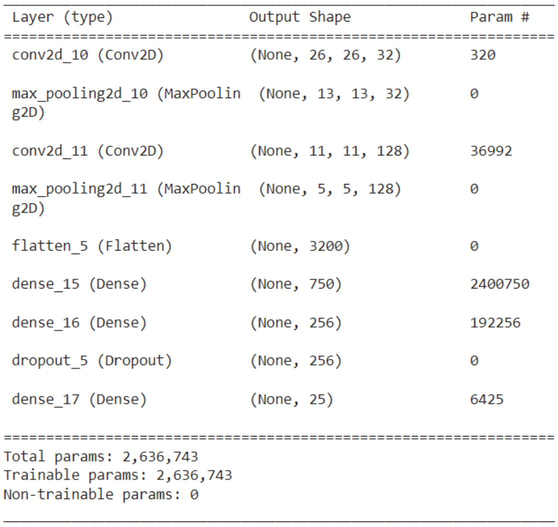
```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_8 (Conv2D)            (None, 26, 26, 32)        320

max_pooling2d_8 (MaxPooling  (None, 13, 13, 32)        0
2D)

conv2d_9 (Conv2D)            (None, 11, 11, 128)       36992

max_pooling2d_9 (MaxPooling  (None, 5, 5, 128)         0
2D)

flatten_4 (Flatten)          (None, 3200)              0

dense_12 (Dense)             (None, 1024)              3277824

dense_13 (Dense)             (None, 256)               262400

dropout_4 (Dropout)          (None, 256)               0

dense_14 (Dense)             (None, 25)                6425

=================================================================
Total params: 3,583,961
Trainable params: 3,583,961
Non-trainable params: 0
```

FIG 5. ARCHITECTURE OF MODEL 3



```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_10 (Conv2D)           (None, 26, 26, 32)        320

max_pooling2d_10 (MaxPoolin  (None, 13, 13, 32)        0
g2D)

conv2d_11 (Conv2D)           (None, 11, 11, 128)       36992

max_pooling2d_11 (MaxPoolin  (None, 5, 5, 128)         0
g2D)

flatten_5 (Flatten)          (None, 3200)              0

dense_15 (Dense)             (None, 750)               2400750

dense_16 (Dense)             (None, 256)               192256

dropout_5 (Dropout)          (None, 256)               0

dense_17 (Dense)             (None, 25)                6425

=================================================================
Total params: 2,636,743
Trainable params: 2,636,743
Non-trainable params: 0
_________________________________________________________________
```

FIG 6. ARCHITECTURE OF MODEL 4



```
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_10 (Conv2D)           (None, 26, 26, 32)        320

max_pooling2d_10 (MaxPoolin  (None, 13, 13, 32)        0
g2D)

conv2d_11 (Conv2D)           (None, 11, 11, 128)       36992

max_pooling2d_11 (MaxPoolin  (None, 5, 5, 128)         0
g2D)

flatten_5 (Flatten)          (None, 3200)              0

dense_15 (Dense)             (None, 750)               2400750

dense_16 (Dense)             (None, 256)               192256

dropout_5 (Dropout)          (None, 256)               0

dense_17 (Dense)             (None, 25)                6425

=================================================================
Total params: 2,636,743
Trainable params: 2,636,743
Non-trainable params: 0
_________________________________________________________________
```

Models 1, 2, and 4 were trained on 10 epochs whereas model 3 was trained on 7 epochs. All models used “sparse_categorical_crossentropy” as the loss function and Models 1, 2, and 3 used a “Stochastic gradient descent (SGD)” optimizer with a learning rate of 0.01, and Model 4 used a “Root Mean Square Propagation (RMSprop)” optimizer with a learning rate of 0.01.

The results of these models were then hard-voted and the class with the highest votes was assigned as an output.

V. RESULTS AND CONCLUSION

In TABLE 1, the accuracy, precision, and recall of individual models and ensemble models can be seen. We can see that the accuracy, precision, and recall have all increased in the ensemble model.

But the metrics are less to be considered a good model.

TABLE 1 METRICS FOR THE MODELS

MODEL	**ACCURACY**	**PRECISION**	**RECALL**
Model 1	0.7822	0.7860	0.7620
Model 2	0.7723	0.7815	0.7408
Model 3	0.7518	0.7660	0.7329
Model 4	0.7534	0.8001	0.7415
Ensemble Model	0.7968	0.8025	0.7775

FIG 7. PRECISION BAR CHART FOR ALL LABELS OF THE ENSEMBLE MODEL

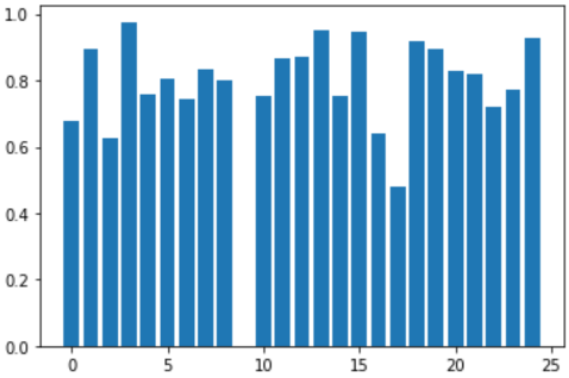
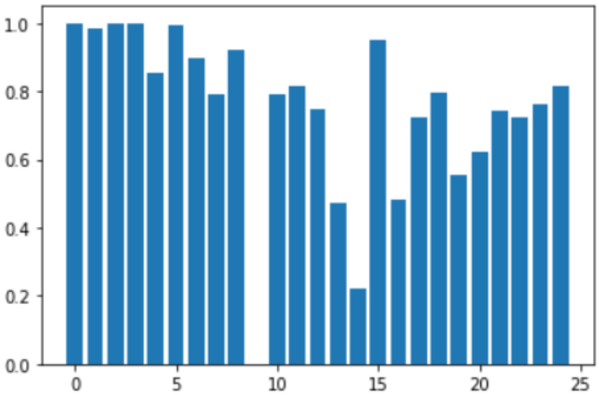


FIG 8. RECALL BAR CHART FOR ALL LABELS OF THE ENSEMBLE MODEL



In FIG 7., it can be observed that precision is overall well distributed except for labels 16 (Q) and 17 (R). In FIG 8., it can be observed that recall of labels 13 (N), 14 (O), 16 (Q), 19 (T), and 20 (U) is very low in comparison with other labels.

VI. LIMITATIONS AND FUTURE WORK

The limitation of this model is the accuracy, precision and recall are less. This could be due to a slight imbalance in the data, and not much variation in the dataset. And the dataset used is a very clean dataset with no disturbance in the background, but real-world data is not always that clean.

To overcome these challenges, data augmentation by scaling, rotating, and even GANS, can be done to increase the variation in data, and oversampling of data for labels with slightly fewer data can be also done. Data can also be oversampled for labels which got less precision and recall, which can further increase the accuracy.

VI. REFERENCES

[1] Yasen, M. and Jusoh, S., 2019. A systematic review on hand gesture recognition techniques, challenges and applications. *PeerJ Computer Science*, *5*, p.e218.

[2] Suarez, J. and Murphy, R.R., 2012, September. Hand gesture recognition with depth images: A review. In *2012 IEEE RO-MAN: the 21st IEEE international symposium on robot and human interactive communication* (pp. 411-417). IEEE.

[3] Li, X., He, Y. and Jing, X., 2019. A survey of deep learning-based human activity recognition in radar. *Remote Sensing*, *11*(9), p.1068.

[4] Adithya, V. and Rajesh, R., 2020. A deep convolutional neural network approach for static hand gesture recognition. *Procedia Computer Science*, *171*, pp.2353-2361.

[5] Singh, D.K., 2021. 3D-CNN based Dynamic Gesture Recognition for Indian Sign Language Modeling. *Procedia Computer Science, 189*, pp.76-83.

[6] Ahuja, M.K. and Singh, A., 2015. Hand gesture recognition using PCA. *International Journal of Computer Science Engineering and Technology (IJCSET)*, *5*(7), pp.267-27.

[7] More, S.P. and Sattar, A., 2016, March. Hand gesture recognition system using image processing. In *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)* (pp. 671-675). IEEE.

[8] Das, S., Biswas, S.K. and Purkayastha, B., 2022. Automated Indian sign language recognition system by fusing deep and handcrafted feature. *Multimedia Tools and Applications*, pp.1-23.

[9] Mannan, A., Abbasi, A., Javed, A.R., Ahsan, A., Gadekallu, T.R. and Xin, Q., 2022. Hypertuned deep convolutional neural network for sign language recognition. *Computational Intelligence and Neuroscience*, *2022*.

[10] Li, W., Shi, P. and Yu, H., 2021. Gesture recognition using surface electromyography and deep learning for prostheses hand: state-of-the-art, challenges, and future. *Frontiers in neuroscience*, *15,* p.621885.

[11] Chakraborty, B.K., Sarma, D., Bhuyan, M.K. and MacDorman, K.F., 2018. Review of constraints on vision-based gesture recognition for human–computer interaction. *IET Computer Vision*, *12*(1), pp.3-15.

[12] Mahmoud, R., Belgacem, S. and Omri, M.N., 2020. Deep signature-based isolated and large scale continuous gesture recognition approach. *Journal of King Saud University-Computer and Information Sciences*.

[13] Dang, T.L., Tran, S.D., Nguyen, T.H., Kim, S. and Monet, N., 2022. An improved hand gesture recognition system using keypoints and hand bounding boxes. *Array*, *16*, p.100251.

[14] Mujahid, A., Awan, M.J., Yasin, A., Mohammed, M.A., Damaševičius, R., Maskeliūnas, R. and Abdulkareem, K.H., 2021. Real-time hand gesture recognition based on deep learning YOLOv3 model. *Applied Sciences*, *11*(9), p.4164.

[15] Kakade, D.N. and Chitode, J.S., 2012. Dynamic Hand Gesture Recognition: A Literature Review. *International Journal of Engineering Research & Technology (IJERT)*, *1*(9), pp.2278-0181.

[16] Simion, G., Gui, V. and Otesteanu, M., 2011, December. A brief review of vision based hand gesture recognition. In *Proceedings of the 10th WSEAS International Conference on Circuits, Systems, Electronics, Control, Signal Processing, and Proceedings of the 7th WSEAS International Conference on Applied and Theoretical Mechanics, CSECS/MECHANICS* (Vol. 11, pp. 181-188).