ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 1, pp: (28-38), Month: January - March 2023, Available at: www.researchpublish.com

Isolated sign language recognition using hidden transfer learning

[1] Kausar Mia, [2] Tariqul Islam, [3] Md Assaduzzaman, [4] Sonjoy Prosad Shaha, [5] Arnab Saha, [6] Md. Abdur Razzak, [7] Angkur Dhar, [8] Teerthanker Sarker, [9] Al Imran Alvy

DOI: https://doi.org/10.5281/zenodo.7615847

Published Date: 07-February-2023

Abstract: **Sign language (SL) is a visual language that people with speech and hearing disabilities use to communicate in their everyday conversations. It is entirely an optical communication language due to its native grammar. Sadly, learning and practicing sign language is not that common in our society; as a result, this research created a prototype for sign language recognition. Hand detection was used to create a system that will serve as a learning tool for sign language beginners. We have created an improved Deep CNN model that can recognize which letter, word, or digit of the American Sign Language (ASL) is being signed from an image of a signing hand. We have extracted the features from the images by using Transfer Learning and building a model using a Deep Convolutional Neural Network or Deep CNN. We have evaluated the proposed model with our custom dataset as well as with an existing dataset. Our improved Deep CNN model gives only a 4.95% error rate. In addition, we have compared the improved Deep CNN model with other traditional methods and here the improved Deep CNN model achieved an accuracy rate of 95% and outperforms the other models.**

Keywords: **Transfer Learning, Reinforcement Learning, Sign Language, Deep Convolutional Neural Networks, VGG16.**

1. INTRODUCTION

Communication is very important to the formation of a nation. Better communication leads to better understanding, which benefits everyone in the community, including the deaf and people with a speech problems. People can communicate with one another using sign language [1]. The majority of hearing people, on the other hand, have no understanding of sign language and learning it is difficult. There is not enough study done yet related to sign language. As a consequence, there is yet a significant gap between the hearing impaired and the hearing majority.

There are different sign languages used by speech and hearing-impaired people from everywhere over the globe, including “American Sign Language, Indian Sign Language, Australian Sign Language, and Italian Sign Language” [2][3]. The major object of our Sign Language Recognition System is to provide a fast and precise way for the hearing-impaired community to recognize sign language more accurately. To get the desired result we built an improved Deep CNN model. To obtain data from the signer, we used Google drive. Our datasets were uploaded there as zip files and we used unzip command in Google Colab to extract them. The system's goal is to serve as a learning tool for those who are interested in learning additional information about the fundamentals of sign language, such as alphabets, words, and digits.

1.1 Motivation

Deaf and hard-of-hearing people can communicate with the rest of society with the help of a good Sign Language Recognition system. The objective of SLR is to create methods and methodologies for correctly recognizing a succession of gestures and understanding their meaning. Sign Language Recognition is a tenacious and motivating task due to numerous constraints and factors. Sign Language Recognition is a notable task because of its impact on society, as speech and hearing-impaired people face a significant communication gap with the speaking community. We wanted to help them to overcome this communication gap and as result, we have come up with this research. There are numerous opportunities to research recognizing American Sign Language to improve communication between the mute and speaking communities. Hand gestures in Sign Language are used in text or speech translation systems or the alternatives for speech and hearing-impaired people that are used in public places like airports, post offices, or hospitals. We have trained our model to identify alphabets, words, and digits. We have worked on this research to help deaf/mute people as well as the hearing society. The author's main objective is to propose an improved Deep CNN that can easily identify sign language, Identify the alphabets or words a person is signing, Identify the digits a person is signing, and reduce the workload and time needed to identify a sign language that can be used in other tasks instead.

1.2 Challenges

Data collection: We have faced so much trouble to customize our data. We had to take all the images for different hand gestures individually. We haven't any idea of sign language hand gestures, or how to express the signs. So, we watched some tutorials and get help by googling how to express sign languages. This took a big amount of time from us.

Data transfer: There are limitations to how much data can be stored from a mobile device as we have taken the images from our mobile. We also need to transfer the images from our mobile to our computer via USB cable. It took a long moment to transfer all the images because we had a big number of images on our mobile and the cable was a mess.

Calculation limitations: There are limitations to how fast data can be processed and hand gestures detected. This can then cause a wrong sign language to be detected. We can rectify this by using cloud computing for our system.

Hardware limitations: There can be bottlenecks in our processing hardware where they will lag behind the captured images. This can easily be rectified by adding more robust hardware like CPU, HDD, or GPU to our architecture. We have faced this problem in our training of the model where we had to leave our computers on for a few days.

Model limitations: From our research, we have seen that some models will underperform in either hand gesture recognition or sign language detection. We have tried to rectify this by using ImageNet and VGG16 models.

Weather and Light: The performance of the system can be severely impacted by adverse weather conditions. There may be limitations on identifying hand gestures in a low light environment like during the evening or night. That's why we had to take the pictures in a bright condition with a clear background.

2. RELATED WORK

The age of information is the current phase of our civilization, and the computer and the internet are the main technological drivers behind it. Deep CNN [14] is a type of specialized neural network that is used to process images. VGG is a model for recognizing objects that can support up to nineteen layers. VGG was created to be a deep CNN, outperforming baselines on a wide range of tasks and datasets outside of ImageNet. The practice of taking the knowledge or features of one problem and applying them to another problem is known as transfer learning [15]. It's like using software to detect cancerous tumor shapes instead of irregular bread shapes. We have already seen a lot of research on Deep Convolutional Neural Networks and Sign Language Recognition. We've looked over a few of these research papers and gotten some ideas from them.

Shivashankara S, Srinath S, et al. [1] proposed an ideal strategy with the main goal of transliterating 24 static American Sign Language alphabets and integers into humanoid or appliance comprehensible English composition. They found out majority voting was the most accurate out of a group consisting of GoogleNet, AlexNet, VGG-16, VGG-19, and ResNet-50. R. Elakkiya, et al. [2] proposed the impact of machine learning in the most up-to-date literature on sign language recognition and classification. This essay primarily focuses on resolving three key SLR concerns. That is to say, elimination and sorting of strong subunit characteristics, stirring epenthesis activity, and fulfilling the appearance for subunit sign modeling.

D. Forsyth, A. Farhadi, and R. White et al. [3] proposed a technique to create word models for American Sign Language (ASL) that can be transferred between signers and aspects. They use transfer learning to build this model. This method is demonstrated in two scenarios, from an avatar to a frontally seen human signer and from an avatar to a 3/4 view human signer. The primary objective of this study is to show the benefits of transfer learning using comparative characteristics. Lean Karlo S. Tolentino, Ronnie O. Serfa Juan et al. [4] proposed a system that converts static sign language into its verbal counterpart, which includes letters, digits, and fundamental phrases static indicators helping people understand the principles of Sign language is a kind of indication that is used to communicate. With regards to testing precision of 90.04 percent in letter acceptance, 93.44 percent in digit identification, and 97.52 percent in static word recognition, this model was capable of reaching 99 percent training accuracy, with a mean of 93.667 percent based on posture recognition having a limited amount of time. A.L.C. Barczak, N.H. Reyes et al. [5] narrated a fresh picture dataset is being created that can be utilized by additional researchers in computer vision. The first report of the training dataset includes 2425 pictures of 5 several people for each of the 36 ASL motions. The future report will have 18000 hands of 20 several people manifested in 5 several ways, with 5 repeats for every one of the 36 ASL motions. The dataset will be continually brought up to date with fresh pictures until it is complete. Md. Shahinur Alam, et al. [6] proposed a model which perceives all 36 letters and 10 digits of the BDSL with consequential accuracy. In the identification and explanation of the BDSL symbol, a sophisticated result with an accuracy rate of 99.57% and a validation loss of 0.56% was achieved.

Teak-Wei Chong, et al. [7] this article describe an American Sign Language recognition system that employs the Controller for Leap allusion with 26 letters and 10 numbers. The study included a total of 23 characteristics, which were then split into six distinct groups of combinations. According to the findings, the space between two fingertips and the neighboring fingers is an important characteristic of sign language understanding. Anna Deza, Danial Hasan, et al. [8] has proposed a neural network model which can determine the given picture of a signing hand of the American Sign Language (ASL) alphabet being signed. This translator would substantially reduce the barrier for many mute people who is unable to hear to converse with others in everyday situations.

J. Pansare and M. Ingle et al. [9] In the field of hand gesture analysis, they proposed two primary approaches, vision-based and device-based. They use Fourier Transformation to build this project. Using the significance of the Binary Linked Object (BLOB) technique, real-time HGRS has been created for the recognition of 26 static hand motions linked to the A-Z alphabets. This model probably recognizes single-hand movements in real time and obtains a detection rate of 90.19%. Cheok, M.J., Omar, Z. & Jaward et al. [10] in this research paper seeks to core on a survey of advanced methods. Although expression on the face is utilized in sign language, it is not covered in this work. These methods are used for pre-processing, analysis, feature extraction, and distribution.S. Upendran and A. Thamizharasi et al. [11] have proposed Deep Learning Computer Vision may be used to detect hand gestures by constructing Deep Neural Network designs (Convolution Neural Network Architectures) in which the model learns to recognize hand gesture photos throughout an epoch. When the model correctly detects the gesture, the matching English text is created, and the text may subsequently be translated to voice. The PCA characteristics in the ASL hand posture are extracted by this method. The collected PCA features may be utilized to categorize the ASL alphabets with the k-NN classifier. Krizhevsky, A., Sutskever, I., Hinton, G.E et al. [12] have proposed ML algorithms are used broadly in ongoing approaches to object recognition. They trained a huge, DCNN to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 competition into 1000 separate frames. On the test data, they attained top 1 and top 5 error rates of 37.5% and 17.0%, respectively, which is roughly superior to the previous advance. Ivan Gruber, Dmitry Ryumin, et al. [13] have proposed a convolutional neural network that may be used to label numerical movements in sign language. They developed a fresh dataset of these motions for the sake of this study. The traditional VGG16 model was used for a distribution job, and the outcome was in comparison to the selected baseline technique and other examined architectures. They achieved recognition accuracy of 86.45%, which is more than 34% better than the specified baseline approach.

2.1 Limitation of Existing Work

After studying the existing project works and papers we discovered that several alternative machine-learning algorithms can be used to perform sign language detection. These existing works have several limitations and we can fix that with our research. For us, the most practical choice is open-source ImageNet, such as VGG16, Transfer learning, and Deep Convolutional Neural networks. We were also required to contribute our dataset for this research. In that instance, our enhanced Deep CNN model will be quite useful in overcoming the constraints.

3. APPROACH AND EXPERIMENTS

3.1 Proposed System

As our research goal is to detect hand gestures and identify sign language, we made use of deep Convolutional Neural networks and Transfer Learning to solve that problem. We made use of DCNN or Deep Convolutional Neural Network algorithms.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 1, pp: (28-38), Month: January - March 2023, Available at: www.researchpublish.com

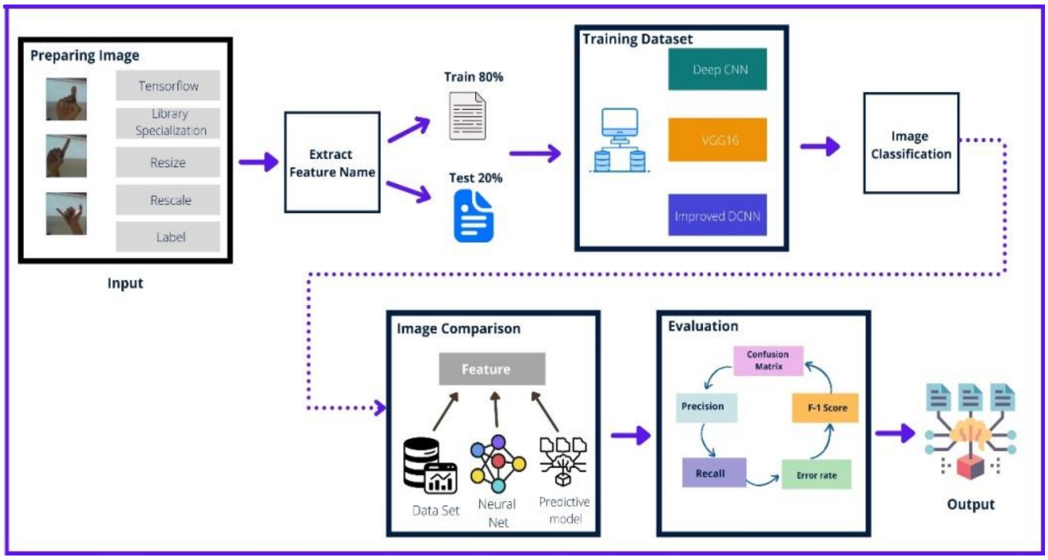

Fig. 1: Working prototype of the proposed methodology

3.2 Dataset

The major source of data for this research was the ASL Alphabet, a collected dataset of American Sign Language (ASL) [16]. The collection is made up of 87,000 200x200 pixel pictures. There are 45 classes in all, 26 for the letters A-Z, 10 for the digits 0-9, and 8 for space, delete, nothing, dislike, ok, good job, home, please, and others. We have also used our custom dataset consisting of 1000 pictures that were taken by our own mobile devices. And later on, we also customized the training dataset with the downloaded dataset. We added 0-9 digits and some words in the training and testing datasets. Our raw images were high-quality images and for that reason, it took a bit long time for loading data.

3.3 Implementation Procedure

3.3.1 Model Training

To develop our Deep CNN model, we have used four Convolution layers and two flattened layers. We used 16, 32, 64,128 filters in the four Convolution layers respectively. In these four layers, we used a kernel size of 2 and the activation function we used here is "Relu" [17]. In the last dense layer, we used the "Softmax" classification algorithm. Here, we trained 15 times with a batch size of 64. In this model, we had 45 classes.

3.3.2 Model Evaluation

In contemplation of assessing our model's performance, we made use of confusion matrices (CM). From CM, we get four basic qualitative model quality indicators, namely, True positive (TP) which is genuinely positive, False positive (FP) which is not genuinely positive, True negative (TN) which is negative in the true sense and False negative which is negative that is not true (FN) [18][19]. Accuracy is a multi-class performance metric to evaluate the model. After comparing accuracy with the VGG16 model and Deep CNN model, our proposed transfer learning-based improved Deep CNN model outperformed the modified Deep CNN model and the VGG16 model. We have created a table named table 1 and we can observe the performance from this table on the dataset which is trained by Deep CNN. Here ten epochs are shown and have five columns Accuracy, Error rate, train loss, valid loss, and time. From this table we get the highest accuracy is 0.9445 and the lowest accuracy is 0.5219.

Table 1: Performance after training with DCNNs

EPOCH	TRAINLESS	VALIDLOSS	ACCURACY	ERROR RATE	TIME
0	1.5251	14.7713	0.5219	0.4781	01:08
1	0.4843	21.1318	0.8332	0.1668	01:07
2	0.2839	25.8563	0.9011	0.0989	01:07
3	0.1973	27.5158	0.9320	0.068	01:07

4	0.1539	33.3173	0.9376	0.0624	01:07
5	0.1294	33.6078	0.9400	0.0600	01:06
6	0.1051	36.5312	0.9412	0.0588	01:06
7	0.0913	35.4029	0.9423	0.0577	01:06
8	0.0878	35.6365	0.9438	0.0562	01:06
9	0.0775	40.9340	0.9445	0.0555	01:07

3.3.3 VGG16

VGG16 is a deep CNN that has 16 layers that's why it is called VGG16 [20]. The most unique feature of VGG16 is that it rather of an enormous number of hyperparameters. It has limited the number of filters in convolution layers to 3x3 with a stride of 1 and used the same padding every time. It also uses the max pool layer of a 2x2 filter and stride 2 [21].

We have created a table named table 2 and from this table, we can see the performance of the dataset which is trained by VGG16. From this table we get the highest accuracy is 0.9113 and the lowest accuracy is 0.5165 in epoch 0 and epoch 9. When our data set is trained with VGG16 a model of transfer learning, there has some loss value we got. The highest train loss happened in epoch 0 which is 1.5251 and the lowest train loss happened in epoch 9 which is 0.0775. The learning complication is recast as a development problem, with a loss function created and the approach was tweaked to reduce the loss function's size. Error, on the other hand, represents how well your network performs on a certain training, testing, and validation set. A low mistake rate is desirable, whereas a large error rate is unquestionably undesirable. A loss function, of which there are numerous, is used to determine the error.

Table 2: Performance after training with VGG16

EPOCH	TRAINLESS	VALIDLOSS	ACCURACY	ERROR RATE	TIME
0	2.2070	4.3844	0.5165	0.4835	18:13
1	1.3427	5.2367	0.7341	0.2659	18:10
2	1.0315	5.9215	0.7959	0.2041	18:11
3	0.8539	6.5137	0.8284	0.1716	18:08
4	0.7345	7.0395	0.8512	0.1488	18:08
5	0.6469	7.5203	0.8712	0.1288	17:53
6	0.5798	7.9606	0.8840	0.1160	17:54
7	0.5260	8.3708	0.8955	0.1045	17:53
8	0.4811	8.7582	0.9066	0.0934	17:56
9	0.4448	9.1174	0.9113	0.0887	17:53

3.3.4 Improved Deep CNN Model

Transfer learning is one kind of process of implementing a preceded learned model to a recent model. For instance, you could utilize the model's expertise to recognize additional things such as sunglasses By transfer learning, we hope to implement what we have accomplished in one activity to improve abstraction in another [22]. The substance learned by a network at "task A" is conveyed to a new "task B." We trained this transfer learning with our dataset. It's produced good accuracy. After that, for our proposed system we used transfer learning to get the appearance from it. We have taken the output from the pre-trained model and added it to our Deep CNN model. It gave a good performance overall as the pre-trained VGG16 model's features transfers into the Deep CNN model [23]. To create the transfer learning network we didn't train the layers of VGG-16 but extracted the features from it and we also add 3 custom layers with it. Here we have used the relu and softmax activation function with a dropout function of 0.2, 0.3. To train our proposed model, we have used a transfer learning methodology with some additional layers to train our improved Deep CNN model. We used the VGG-16 model to extract the features and added our Deep CNN model with it to get better output results.

We produced a table called Table 3 and from it, we can examine the performance of a dataset trained using Improved Deep CNN. From this table, we get the highest accuracy is 0.95 and the lowest accuracy is 0.87 in epoch 0 and epoch 9. When our data set is trained with Improved Deep CNN, there have some loss values we got. The highest train loss happened in epoch 0 which is 0.3687 and the lowest train loss happened in epoch 9 which is 0.0912. Error, on the other hand, represents how well your network performs on a certain training, testing, and validation set. A low mistake rate is desirable, whereas a large error rate is unquestionably undesirable. A loss function, of which there are numerous, is used to determine the error.

Table 3: Performance of Improved Deep CNN

Epoch	Train loss	Valid loss	Accuracy	Error rate	Time
0	0.3687	8.2469	0.8773	0.1227	18:13
1	0.2944	11.4947	0.9027	0.0973	18:10
2	0.2473	14.3312	0.9172	0.0828	18:11
3	0.2186	16.3780	0.9254	0.0746	18:08
4	0.1940	17.7813	0.9361	0.0639	18:08
5	0.1695	29.9138	0.9445	0.0555	17:53
6	0.1580	20.6153	0.9480	0.0520	17:54
7	0.1391	22.2125	0.9494	0.0506	17:53
8	0.1318	23.1417	0.9500	0.0500	17:56
9	0.0912	24.9417	0.9505	0.0495	17:53

4. RESULTS AND DISCUSSIONS

4.1 Parameter Optimization

During the simulation of our proposed model, we concentrated on a few parameters to improve its performance. Grow up the batch size to 64 for that It increased our model performance. Though we used a learning rate between 1e-5 and 1e-4, it didn't improve our model performance. Even after trying up to 20 epochs, it didn't improve our performance that much. So, we used 10 epochs. In the case of the activation function, we have used 'ReLU' and we used 'Softmax' as a classification algorithm which is also an activation function that reduced training loss. The main anticipating analytics like accuracy, recall, specificity, and precision are shown by applying confusion matrices. Confusion matrices are important because they let you analyze values like positives that are true, positives that aren't true, true negatives, and negatives that aren't true in a simple way.

Here in Figure 2,3,4, the confusion matrix of the Deep CNN, VGG16, Improved Deep CNN model after training the model with our given images. The diagonal part indicates the number of images it has been detected correctly.

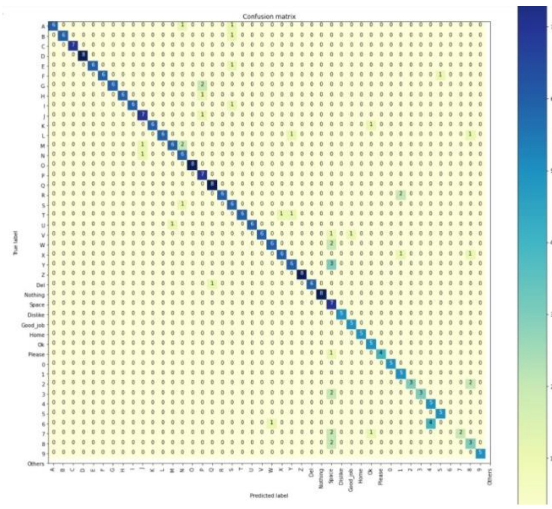

Fig. 2: Confusion matrix of Deep CNN

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 1, pp: (28-38), Month: January - March 2023, Available at: www.researchpublish.com

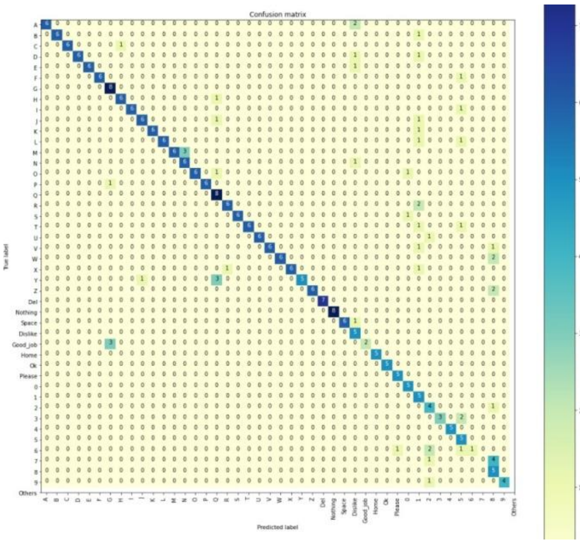

Fig. 3: Confusion matrix of VGG16

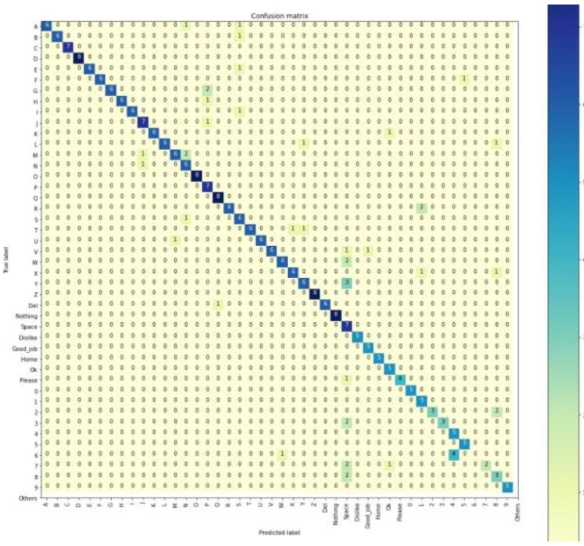

Fig. 4: Confusion matrix of Improved Deep CNN

4.2 Performance Analysis

Here, our proposed transfer learning-based Deep CNN model outperformed the general DCNN model and the pre-trained VGG16 model. This Deep CNN model is easy to implement in Tensorflow and Keras frameworks. It also can detect features from images automatically which makes it so easy to use. If we analyze the 3 different models we use, the performance of the transfer learning-based Deep CNN model is very good. For multiclass image classification, Deep CNN is the best model. It also takes less time to run the procedure which is very important for a model like this.

The table we have created here is one of the different models we have tested on our 45 classes. Here we measure precision, recall, f1-score, and support. Precision, as common as a high probability of success, is the proportion of incidences that are applicable among the recovered. The scores assigned to each class indicate the classifier's accuracy in categorizing data points in that class when compared to all other classes. The number of samples of the real answer that fall into that class constitutes the support. From this table, we understand very quickly which model is better than another model by measuring all model precision, recall, f1-score, and support. Here we show the pixel intensity histogram of an image from our training dataset.

ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 11, Issue 1, pp: (28-38), Month: January - March 2023, Available at: www.researchpublish.com

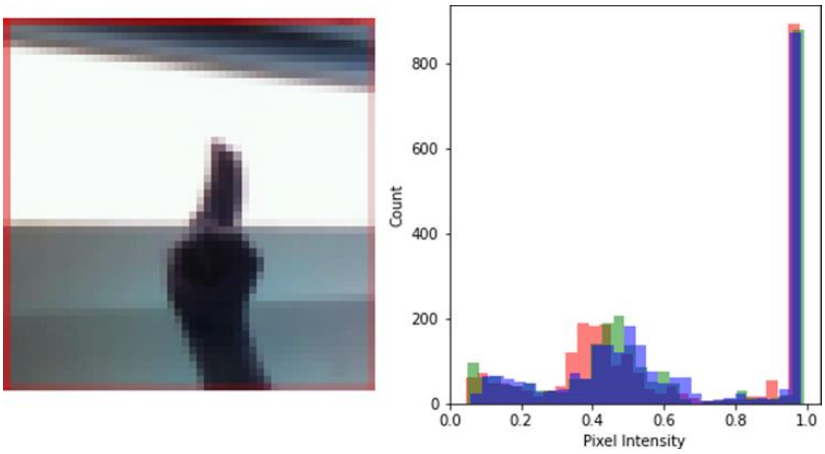

Fig. 5: Pixel Intensity Histogram

Here we have shown a sign language that represents the alphabet "E" and it is correctly defined.

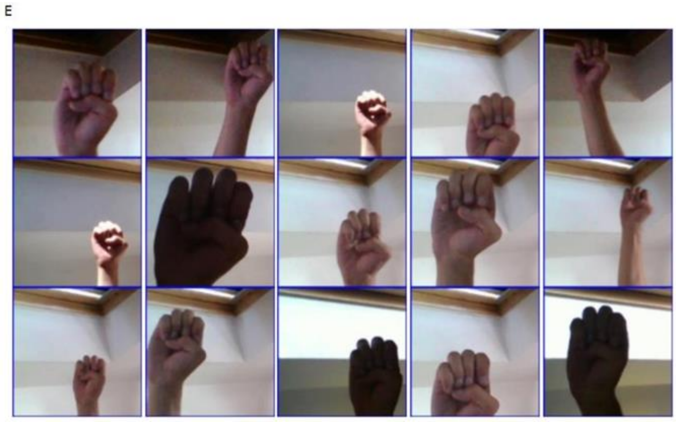

Fig. 6: Character representation of "E"

On the test data, 93.57% accuracy was attained using the suggested model. In the relevant field, it improves by 2.43% over the previous high of 95%. The suggested model has a validation loss of 8.24%. The model has been created and trained on the arranged dataset in this manner that it provides almost accurate results. The performance graphs give a clearer picture of the outcomes. The vertical and horizontal axes have been rescaled by 1/100 and 1/10, respectively. The horizontal axis indicates the number of epochs, while the vertical axis is the percent accuracy or loss. The suggested model's performance accuracy curve is shown in Figure 7. The proposed model's highest validation accuracy has been determined to be 99.57%. For increasing epochs, no additional suggestive improvement was found for the proposed model.

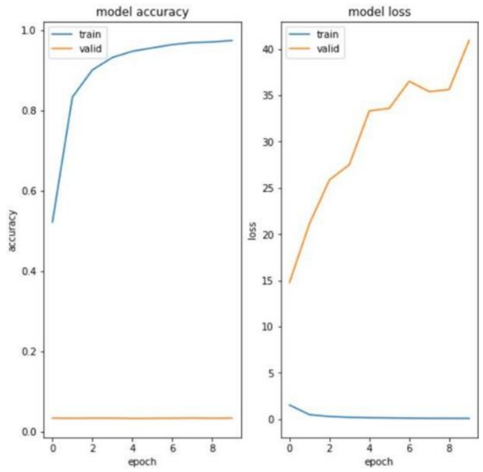

Fig. 7: The learning curve of the custom CNN model.

The validation loss and accuracy curve of the proposed model is shown in Figure 8. The validation loss has been shown to saturate at 0.56%. A comparison chart showing the accuracies of comparable works provides a realistic picture of performance improvements. The suggested model in this study recognizes all 46 Sign Language characters with high accuracy. On the test data provided by outsiders, the proposed model obtains a 99.41% accuracy. For the test data in this scenario, the validation loss for the suggested model is 0.70%.

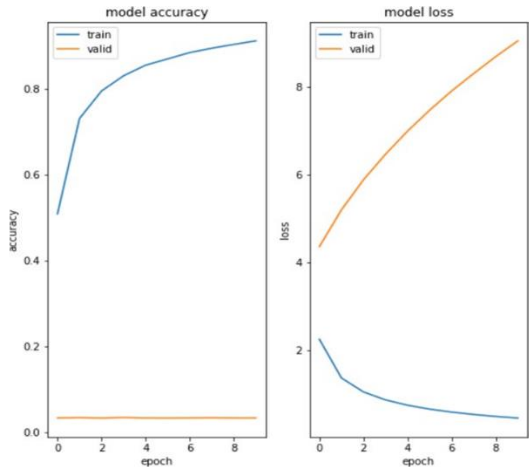

Fig. 8: The learning curve of the VGG16 model.

Figure 9 shows the Improved Deep CNN model's performance accuracy and loss curves. The model was used to predict the characters of BSL in various lighting circumstances, and the model was used to extract the shortest transition time between characters. The suggested model was used to predict characters at various angular deviations of certain hand signals, and the related probability distribution of the characters was recovered from the model. Up to 30 degrees of angular variation, the model gives significant predictability for the hand signals of SLC.

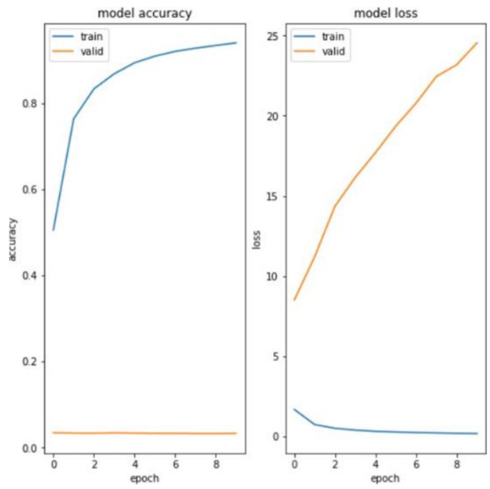

Fig. 9: The learning curve of the Improved Deep CNN model.

Comparative analysis means comparing our model with other traditional ones. After comparing accuracy with the VGG16 model and a general Deep CNN model.

5. CONCLUSION AND FUTURE WORK

This research study proposed a hand gesture detection and sign language recognition system using Deep CNN with VGG16 of ImageNet. In terms of detecting hand gestures, it gave good results with Deep CNN and VGG16. When we used transfer learning our Improved Deep CNN model gave better results than the other methods. Moreover, it gave 95% accuracy over all the classes. Here we have used image data of many hand gestures. Which we hope will be helpful for speech and hearing-impaired people in the future. In our current model, we cannot detect hand gestures from real-time data or video data. In the future, we will build a model which will be able to detect hand gestures from real-time or video data. Also, we will increase the variations of the hand gestures and more words so that it can detect more words from different angles.

REFERENCES

[1] Shivashankara S, Srinath S, " American Sign Language Recognition System: An Optimal Approach ", International Journal of Image, Graphics and Signal Processing(IJIGSP), Vol.10, No.8, pp. 18-30, 2018. DOI: 10.5815/ijigsp. 2018.08.03

[2] R. Elakkiya "Machine learning based sign language recognition: a review and its research frontier" Journal of Ambient Intelligence and Humanized Computing https://doi.org/10.1007/s12652-020- 02396-y Received: 23 April 2020 / Accepted: 23 July 2020 © Springer-Verlag GmbH Germany, part of Springer Nature 2020.

[3] D. Forsyth, A. Farhadi, and R. White, "Transfer Learning in Sign language," in 2007 IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, 2007 pp. 1-8. DOI: 10.1109/CVPR.2007.383346

[4] Lean Karlo S. Tolentino, Ronnie O. Serfa Juan, August C. Thio-ac, Maria Abigail B. Pa-mahoy, Joni Rose R. Forteza, and Xavier Jet O. Garcia "Static Sign Language Recognition Using Deep Learning" International Journal of Machine Learning and Computing, Vol. 9, No. 6, December 2019.

[5] A.L.C. Barczak, N.H. Reyes, M. Abastillas, A. Piccio and T. Susnjak "A New 2D Static Hand Gesture Colour Image Dataset for ASL Gestures" Res. Lett. Inf. Math. Sci., 2011, Vol. 15, pp. 12–20 Available online at http://iims.massey. ac.nz/research/letters/.

[6] Md. Shahinur Alam, Mahib Tanvir, Dip Kumar Saha, Sajal K. Das "Two Dimensional Con-volutional Neural Network Approach for Real-Time Bangla Sign Language Characters Recognition and Translation" Received: 16 February 2021 / Accepted: 18 July 2021 The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd 2021.

[7] Teak-Wei Chong, Boon-Giin Lee "American Sign Language Recognition Using Leap Mo-tion Controller with Machine Learning Approach" Received: 11 September 2018; Accepted: 17 October 2018; Published: 19 October 2018.

[8] Deza, Anna, and Danial Hasan. "MIE324 Final Report: Sign Language Recognition." on 2nd December 2018.

[9] J. Pansare and M. Ingle, "Vision-Based Approach for American Sign Language Recognition Using Edge Orientation Histogram," in 2016 International Conference on Image, Vision, and Computing, 2016, pp. 86–90.

[10] Cheok, M.J., Omar, Z. & Jaward, M.H. "A review of hand gesture and sign language recognition techniques". Int. J. Mach. Learn. & Cyber. 10, 131–153 (2019). https://doi.org/10.1007/s13042-017- 0705-5

[11] S. Upendran and A. Thamizharasi, "American Sign Language interpreter system for deaf and dumb individuals," 2014 International Conference on Control, Instrumentation, Com-munication and Computational Technologies (ICCICCT), 2014, pp. 1477-1481, doi: 10.1109/ICCICCT.2014.6993193.

[12] Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)

[13] Ivan Gruber, Dmitry Ryumin, Marek Hr´uz, and Alexey Karpov "Sign Language Numeral Gestures Recognition Using Convolutional Neural Network".

[14] Barbhuiya, A.A., Karsh, R.K. & Jain, R. "CNN-based feature extraction and classification for sign language". Multimed Tools Appl 80, 3051–3069 (2021), Published: 18 September 2020 https://doi.org/10.1007/s11042-020-09829-y

[15] S. J. Pan and Q. Yang, " A Survey on Transfer Learning," in IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 10, pp. 1345-1359, Oct. 2010

[16] Ahmed Khan. Sign Language Gesture Images Dataset. URL: kaggle.com/ahmedkhanak199 5/sign-language-gesture-images-dataset (ac-cessed: 03.01.2022).

[17] Demir, Fatih, Daban Abdulsalam Abdullah, and Abdulkadir Sengur. "A new deep CNN model for environmental sound classification." IEEE Access 8 (2020): 66529-66537.

[18] Pineda-Jaramillo, Juan D. "A review of Machine Learning (ML) algorithms used for modeling travel mode choice." Dyna 86.211 (2019): 32-41.

[19] Cho, Gyeongcheol, et al. "Review of machine learning algorithms for diagnosing mental illness." Psychiatry investigation 16.4 (2019): 262.

[20] Theckedath, Dhananjay, and R. R. Sedamkar. "Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks." SN Computer Science 1.2 (2020): 1-7.

[21] Rezende, Edmar, et al. "Malicious software classification using VGG16 deep neural network's bottleneck features." Information technology-new generations. Springer, Cham, 2018. 51-59.

[22] Xia, Kai-jian, Hong-sheng Yin, and Jiang-qiang Wang. "A novel improved deep convolutional neural network model for medical image fusion." Cluster Computing 22.1 (2019): 1515-1527.

[23] Hu, Gensheng, et al. "Identification of tea leaf diseases by using an improved deep convolutional neural network." Sustainable Computing: Informatics and Systems 24 (2019): 100353.