Gesture Recognition for Interpretation of Bengali Sign Language using Hyper Parameter Tuning Convolution Neural Network

Tahmina Akter
Department of Computer Science and Engineering
Port City International University
Chattogram, Bangladesh

Muhammad Anwarul Azim
Department of Computer Science and Engineering
University of Chittagong,
Chattogram, Bangladesh

Mohammad Khairul Islam
Department of Computer Science and Engineering University of Chittagong, Chattogram, Bangladesh

ABSTRACT
In recent years, the proportion of deaf and dumb persons has increased dramatically all across the world. Physically challenged persons have found it challenging to communicate with normal others. Via the movement of gestures like the face, hand, and body, communication takes place. The goal of this research is to develop a hand gesture recognition method that can understand sign language for deaf and hard of hearing persons. We collected our hand sign data from a well-known source called Ishara- Lipi. Because of the low amount of gestures, we augment this data set from 1009 images to 9360 images for our recognition purposes. This augmented dataset is divided into the following sections: training and testing. We develop the Convolutional Neural Network (CNN) model that uses multilayer CNN, followed by pooling layers and dropout layers and also adding multiple hidden layers, or simply called dense layers afterward. We experiment with our CNN model with the tuning of hyper parameters in all possible combinations. Our model provides 98.78 percent and 97.45 percent accuracy in training and validation respectively. We evaluate our trained CNN model through the test dataset that also has provided both 98 percent for accuracy and f1 score.

General Terms
Image Processing, hand gesture Recognition.

Keywords
Gesture recognition, sign language, CNN, hyper parameter tuning.

1. INTRODUCTION
Speech impaired individuals are usually deprived of normal communication in society with other individuals. Bangladesh has 150 million populations, of which 1.5 million are hearing impaired. To aid people in this community a recognition system of hand gestures would be quite helpful.

1.1 Background
Due to Birth defects, accidents, and oral disorders are the primary reason behind growth in the number of deaf and dumb people in recent years. Per the 2001 Disability Welfare Act, speech impairment is characterized as loss of one's ability to utter/pronounce meaningful vocabulary sounds, or impaired, partially or entirely as well as not operating normally (Ahmed and Akhand 2016). Interacting with normal people has been tough for this community. They are frequently unable to communicate their message to others. The regular community, as a result, remains unhappy and apprehensive under that situation. It is very important to make these people part of society in this era of technology by helping them communicate smoothly. The most important means of communication is Sign Language for the deaf and dumb community.

1.2 Sign language
Sign language (also known as signed language) is a language that uses manual communication to convey messages to others. To convey a speaker's thoughts by using hand gestures, movement, finger, arm, or body orientation, and facial expressions all at the same time. For persons who are unable to talk or hear, sign language is one of the most useful communication methods. It is a boon for deaf and dumb individuals. It is basically a collection of numerous motions, shapes, and movements involving the hand, body, and face. Deaf people utilize this unique gesture to express sentiments and thoughts that they are unable to express verbally. Every hand gesture, facial expression, and body movement has its own distinct meaning (Chen et al., 2013). Different types of sign languages are used in different regions of the world. Sign language in each region is determined by the spoken language and culture of that region; for example, American Sign Language (ASL) is used in the United States, while British Sign Language (BSL) is used in England and also Japanese sign language, French sign language (FSL), and Indian sign language (Chen et al., 2019). Bangladesh has just established a formal sign language. In 2000, an initiative was taken by the Centre for Disability in Development (CDD) to standardize communication with sign languages in our country. There were different local variants before this step, and no national dialect existed in their training center. People in Bangladesh are still unaware of this kind of interaction, so deaf community can still not lead a simple life like ordinary people (Hoque et al., 2016).

1.3 Image processing
Image processing is one of the most rapidly developing branches of computer vision, with applications in a wide range of fields. It has the potential to develop the optimum machines in the future that can mimic the visual function of living beings, for example; it is the foundation of all types of visual automation. (Trigueiros et al., 2014; Wang and Xi, 1997).

1.4 Application Area
Sign language is not a widespread language. Pattern recognition, computer vision, natural language processing, and psychology involved in Sign language recognition is a multidisciplinary research field (Itkarkar and Nandi, 2013). There is enormous application of hand gesture such as virtual Environment control, video surveillance, robot remote control. It's also employed in news broadcasting and billboard

advertising (Khan et al., 2020). It is also employed in the world of musical composition. It also has a major use in sign language translation. For sign language translation, it is usually divided into two components. The first module converts Standard English sentences to SL (for deaf people to understand), and the second module converts SL to English text (to be understood by normal people). The first module is not required for informed hearing impaired people who can read English. Both modules are necessary for illiterate deaf and dumb persons. A language processing engine that is based on precise language commands is necessary in both modules. Pattern recognition and language processing, as well as computer vision and image processing, as well as linguistic research, are all involved in the conversion of sign to text (Huong et al., 2015).

1.5 Contribution

In this research, we present a vision-based technique for recognizing hand gestures. For this, we used a deep learning technique that involves using the Convolutional Neural Network (CNN) to train the classifier. Furthermore, we used test gestures as input to the learned CNN model to determine recognition accuracy. During training and validation, our model is 98.78 percent and 97.45 percent accuracy, respectively. We also tested this trained model using a test dataset, which provided both 98 percent accuracy and f1 score.

2. LITERATURE REVIEW

A method that noted hand movements of alphabetic Vietnamese Sign Language (VSL) characters using a series of video images taken from the depth sensor in a Microsoft Kinect. Microsoft Kinect is nothing; however, as long as Kinect focal points and sights are spaced, there is a depth sensor that arranges depth images. They used key frames for object extraction. In addition, the Support Vector Machine also performed (SVM) on their method. 91 percent accuracy is given by the suggested method (SHINDE and AUTEE, 2016).

A system recognizes the hand gestures using the calculation of peak and angle of the hand movements. For their system, they captured the color images using a webcam that has an intensity of 20 megapixels. After that, they performed image preprocessing operations such as RGB to gray conversion, background segmentation, and noise reduction. They used computation of angle and peak for extracting the hand feature from the segmented hand picture, as well as edge detection, Gabor filter, and Hidden Markov model for additional extracting features. For peak calculation that count the raised and folded fingers from the given images. These angles are distinguished into three categories. Moreover, these zero, positive, and negative angles are converted into the 12-bit generation of a binary sequence.Lastly, there are MATLAB built-in commands for converting motions to voice, including zero, one, negative. They worked on converting hand movements into sounds and vice versa in their system (Dabre and Dholay, 2014).

A visual recognition system for Indian Sign Language was suggested, which translated gesture phrases into text and voice. The working approach of their suggested system included two major phases: image preprocessing and classification. They use more than 5 approaches in the image preprocessing phase, including background subtraction, blob analysis, RGB to gray conversion, filtering, brightness/ contrast adjustment, and scaling. They employed three categories of samples in the classification stages: positive, negative, and test samples. For text results, they use the Haar Cascade Classifier on the sample dataset. Using the “System Speech Synthesis” library in Microsoft. Net framework translating text information to audio (Saha et al., 2018).

They were suggested that the hand gesture recognition method be improved with acceptable precision, with the hand providing the input to the pattern recognition system. ASL, or American Sign Language, is one instance of a possible reference model. The image is preprocessed after being captured using a webcam. Further segmentation of the image is done using polygon approximation and approximate convex decomposition. The feature extraction process is completed by capturing the unique feature among the hand's many convex segments. The obtained singularities are then used to produce the extracted feature vectors (Kishore et al. 2016).

For gesture categorization, back propagation neural networks were used to extract hand tracks and hand shape attributes from continuous sign language images. The tracking feature and shape features were extracted using two separate methods: Horn Schunck optical flow (HSOF) and Active Contours (AC) (Prakash et al.,2017).

A different gesture recognition and fingertip detection algorithm for human-computer interaction has been developed for specific mouse control operations through a real-time camera. The camera captures the sign languages in real time. The collecting region examined by morphological operations is a segmented, increasing algorithm of the hand area alone. The centroid of the palm region is calculated, and the fingertips are then identified using the convex hull technique (Gani and Kika, 2016).

A real-time gesture system recognizes Albanian Sign Language. They employed a Kinect device to capture the hand gestures in their recognition system. Following the capture of hand images, hand segmentation is carried out by defining a constant threshold value in each pixel of the image's data. The K-means clustering technique is used to categorize hands and divide pixels into two groups. The centroid distance is taken into account after extracting the hand's contour pixels, and Fourier descriptors for images of the hand form are created. The minimum Euclidean distance for each input gesture associated to the training data set is obtained by calculating Fourier coefficients and Euclidean distance at a specific time (Dong et al., 2015).

3. METHODOLOGY

The proposed approach is provided in this section. Figure 1 depicts a high-level overview of the proposed strategy. Image data was acquired from data sources and data augmentation was performed. The augmented data was then divided into two modules: training and test datasets. We also used the training dataset to build our CNN model. A test dataset has also been applied to this model. As a result, we examined the test model's performance. In addition, the proposed technique has been separated into many parts.

3.1 Image Acquisition

Image acquisition refers to the acquisition of images using digital media such as a digital camera or webcam, as well as obtaining images from many types of sources erez-Sanz et al., 2017). For our suggested system, we obtained an image dataset known as the Ishara-lipi dataset. In this Ishara-lipi dataset, there are 50 sets of 36 Bengali sign symbols with 6 vowels and 30 consonants that express the message of all Bengali letters. There are 1800 photos in total in this dataset (Hussain et al., 2017).

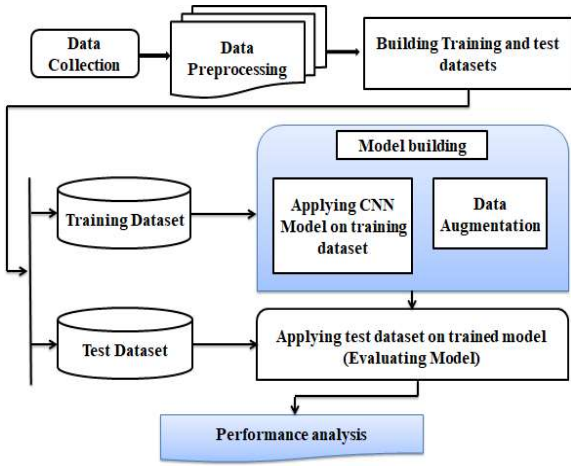

Figure 1 The overview of proposed method.

3.2 Build Train and Test dataset

We proceeded by preprocessing the data we had gathered. With the help of data augmentation, we were able to increase the size of the existing dataset (Wei and Zou (2019), ogelsang and Erickson (2020)). We then split the entire dataset for training and tests. There are 7,200 images in 36 classes in our training dataset. There are 200 images in each class. In test dataset, there are 60 images in each class. In test dataset, there are 60 images in each 36 classes.

3.3 Build Model on Train data set

To train the data set, we'll need to build a fully convolutional Neural Network (CNN) model. There are numerous types of classes or data types in today's dataset. Deep learning is the best approach and the most popular subset of machine learning in this scenario due to its high level of accuracy. Building a Convolutional Neural Network for image classification is a wonderful approach to apply deep learning (Albawi et al., 2017).

3.4 Machine Learning and Deep Learning

In traditional machine learning, there are multiple stages, including data collection and feature generation in the first stage. Preprocessing and feature selection are included in the second stage. Data partitioning, learning, and assessment are the third steps of machine learning.One of the most often utilized techniques in computer vision-based applications is the CNN framework. CNN has excellent capabilities for feature generation and classification Xu et al. (2020).

3.5 Convolution Layer

The convolution layer is one of the fundamental and most important layers of a convolutional neural network. Its goal is to use convolution filtering to detect the existence of a collection of features from input images in the input layer (Alani et al., 2018).

Convolution kernels or filters are designed to extract local features from the input. The idea of convolution filtering is to "slide" a window representing the image feature and perform the convolution product between the feature and each portion of the scanned image, followed by a non-linear activation function. The purpose of convolution operation is to extract low-level features such as edges, color, gradient orientation, etc., and also high-level features by adding more convolution layers in the CNN model that is done on our model (Hussain et al., 2017).

The dimension of our input image is 64 (Height)* 64 (weight) * 3 (number of channels). The convolution kernel or convolution filter has been selected as a 3*3*64 matrix followed by the non-linear activation function "ReLU". Extracting high-level features from the first convo layer, we apply two more convolution layers with the "same padding" operation that the dimensionality is either increased or remains the same. In our second and third convolution block, we augment our input image to find that the convolved matrix turns out to be of the dimensions 64*64*3 followed by non-linear activation function "ReLU" and also "same padding" operation.

3.6 Pooling Layer

The pooling layer, also known as the down-sampling layer, is useful in decreasing the resolution of earlier feature maps produced by convolution layer (Zhang et al., 2019). It accepts many feature maps for extracting the dominant features that are positional and rotational invariant to use a pooling layer between two convolution layer (Hahn and Choi, 2020). Furthermore, a pooling operation reduces the dimensionality by cutting the inputs into disjoint parts with the use of the kernel and produces maximum or average output from each region.

3.7 Dropout Layer

There are many regularization methods for neural networks such as data augmentation, weight decay, batch normalization, and dropout. Dropout is a regularization technique used to prevent over fitting from a model. Using a certain probability (0.2, 0.3, and 0.5), dropout increases the generalization of the model by randomly skipping some units or connections (Xu et al.,2019).

3.8 Fully Connected Layer

The fully connected layer is extensively used for classifying at the network's end. It categorizes the images as network input. It accepts data from the previous layers and produces a set of vectors representing the number of categories in our classification problem (Wang et al., 2019).

3.9 Tuning the hyper-parameter of the Layers

The model architecture of one convolution neural network differs from another using hyper-parameter tuning. Tuning of hyper-parameter that is most important for both convolutions and pooling layer's output feature maps (ibaeva, 2018). In our model, we also tuning of hyper-parameters such as number of CNN layer selection and filters, activation function in CNN layer, dropout in CNN layers, batch size selection, number of dense layers, the dropout rate in a dense layer, activation function in a dense layer, optimizer selection, train and validation spilt selection, and number of the epoch.

3.10 Layers and Filters Selection

In convolution operation, the convolutional layers are only convolved on input data such as raw pixels, but it also is applied on the output of another convolutional layer. In general, filters of a first convolutional layer that operate on input data will extract low-level features from input data e.g. lines. The filters of the second convolutional layer, which operate on the output of the first convolutional layers, may extract features that are combinations of lower- level features, such as features that are made up of several lines to depict shapes. More convolution layers are added until almost the deep stages extract high-level features such as faces, houses, and animals (Kligvasser et al., 2018).

3.11 Activation Function

Activation functions are also called mathematical equations that determine the output of a neural network. The activation function is connected to each neuron in the network and determines whether it should be active ("fired") or not for the model's prediction, based on whether the input of each neuron is relevant (Kligvasser et al., 2018). The activation function also serves as a mathematical "gate" between the current neuron's input and its output, which is sent to the next layer. It can be as simple as a step function that turns the neuron output on and off based on a rule or threshold. It can also be a transition that converts input signals into output signals that the neural network requires to function (Adithya and Rajesh, 2020). Non-linear activation functions are used in today's neural network models. They allow the model to produce sophisticated mappings between the network's inputs and outputs, which is useful for learning and modeling non-linear or high-dimensional data, such as images, video, audio, and data. Non-linear activation functions include the sigmoid / logistic, tanH / hyperbolic Tangent, ReLU (Rectified Linear Unit), Exponential Linear Units (ELUs), and Selu functions.

3.12 Optimizer

The basic goal of machine learning and deep learning, which is sometimes referred to as a Cost function or Loss function, is to reduce the gap between both the predicted and actual output. Choosing an optimizer is the ideal technique to pull and change the parameters (weights) of a model during the learning phase in order to attempt and minimize that loss function and make model predictions as accurate and optimized as feasible (Ozcan and Basturk, 2019). We use a variety of optimizers to improve model prediction, including Adagrad, Adamax, Nadam, Ftrl, SGD, RMSprop, Adam, and Adadelta.

3.13 Train vs. Validation Split

The selection of splitting train and validation ratio is also an important part of a model for achieving good accuracy. The entire train dataset was separated into two parts: the train dataset and the validation dataset. The accuracy of learning models has a wide range focused on data set's splitting mechanism.

3.14 Number of Epochs

The iteration number of epoch is a type of hyper- parameter used in machine learning that specifies how many passes through the learning model or algorithm the full training data set has completed. It aids in the enhancement of the learning model's performance by reforming the internal model's parameters. Furthermore, it is minimizing the learning model's failure rate in order to pass a high number of epochs Ozcan and Basturk (2019).

3.15 Performance Analysis

Understanding of performance evaluation matrices of a classification based deep convolutional neural model, it is best practice to generate classification report on the trained classification model. Classification report is the representation of precision, recall, F1 score, accuracy, and support for the model. It's indicates the classification matrices on each per class. This demonstrates a more clear illustration of the classifier in the model, showing that the train classifier model can determine if a class has been highly recognized or not in the test dataset. Making a classification report based on a test dataset is a good practice. One can easily visualize the classification matrices such as accuracy, precision, recall, F1 score, and also number of support used in per class. Furthermore, we may evaluate how much TP, TN, FP, and FN predictions our classifier model makes based on test data using this classification report.

4. EXPERIMENTAL SETUP and RESULT

In this paragraph, we have discussed our experimental setup based on hyper parameter tuning and shown the F1 score as a result.

4.1 Layers and Filters Selection in CNN layer

In our model, we have experimented adding five convolutional layers with different filters. In this experiment, our trained CNN model is good worked at third convolutional layers with different filters such as 64, 128, 256, and 512. In fourth and fifth convo layer, our model f1score has decreased dramatically.

4.2 Activation Function in CNN Layer

We have Applied the non-linear activation functions such as ReLU, tanh, Elu, and Selu in our model, for making assumptions which non-linear activation functions work sounds well. We have showed the accuracy Table 1 on applying different non-linear activation functions. ReLU is provided the highest F1 score among all others activation functions.

4.3 Dropout in CNN Layers

We improved the performance of our trained model by adding dropout probabilities of 0.1, 0.2, 0.25, 0.3, 0.4, and 0.5, as shown in Table 1. Our model predicted a good F1 score of 61.94 percent, with a 0.2 percent dropout rate.

4.4 Number of Dense Layers

We have added four hidden layers in our model for improvement of model accuracy and checked which layer is fitted. Our model is given 58.47 percent F1 score in layer 2 among 4 different layers which show Table 2.

4.5 Dropout within Dense Layers

In our fully connected layer, we have used a set dropout rate. After applying the dropout rate in our model, the percentages of the F1 score are provided in Table 2.

Table 1 Dropout within & Activation Function in CNN Layer

Dropout Rate	F1 Score	Activation function	F1 Score
0.1	61.8	**ReLU**	**55.50**
0.2	**61.94**	tanh	40.72
0.25	53.75	Elu	44.16
0.3	44.02	Selu	39.16
0.4	35.97		
0.5	44.16		

Table 2 Number of Dense Layers & Dropout rate in Dense Layers

No. of Dense Layers	F1 Score	Dropout Rate	F1 Score
Layer-1	57.22	0.05	52.36
Layer-2	**58.47**	0.1	54.72
Layer-3	55.13	**0.2**	**56.38**

Layer-4	53.19	0.3	41.11
		0.4	8.33
		0.5	34.45

4.6 Activation Function in Dense Layer

The various types of non-linear activation functions that apply to the convolutional layer are addressed. In our fully connected layer or dense layer, we use this different non-linear activation function once more. The F1 score of using a non-linear function in dense layers is shown in Table 3.

4.7 Performance based on Optimizer

For model prediction, we use a variety of optimizers e.g. Adagrad, Adamax, Nadam, Ftrl, SGD, RMSprop, Adam, and Adadelta. We show the accuracy Table 3 which the optimizer gives the highest accuracy.

Table 3 Activation Function and Optimizer in Dense Layer

Activation Function	F1 Score	Optimizers	F1 Score
ReLU	**72.08**	SGD	24.72
tanh	8.33	RMSprop	63.88
Elu	54.16	Adam	49.02
Selu	58.19	Adadelta	9.58
		Adagrad	18.47
		Adamax	43.88
		Nadam	**70.97**
		Ftrl	8.33

4.8 Final Activation Function in output layer

In our output layer, we used non-linear activation functions such as Sigmoid, Softmax, and Softplus, all of which provide good accuracy. Table 4 shows that Softplus activation function has the highest F1 score of 67.63 percent among all activation functions.

4.9 Performance based on Train vs. Validation Split

In our model, we fragmented the train dataset in different ratio such as (60 to 40), (70 to 30), (80 to 20), and (90 to 10). In (90 to 10) split ration is given 76.94 percent F1 score among others spit ratio which shows Table 4.

Table 4 Final Activation Function &Performance based on Train-Validation Split.

Activation Function	F1 Score	Split Ratio	F1 Score
Sigmoid	49.027	60 – 40	59.58
Softmax	58.055	70 – 30	54.44
Softplus	**67.63**	80 – 20	64.44
		90 – 10	**76.94**

4.10 Number of Epochs

There are 700 epochs in our trained CNN model that are rapidly running out. Table 5 displays the training accuracy, validation accuracy, F1 score, and accuracy for classes (1-36).

Table 5 Performance on Epoch

No. of Epochs	Train accuracy	Validation accuracy	F1 Score	Accuracy
100	86.67	92.36	90.65	90.69
200	91.21	95.69	93.67	93.65
300	92.57	95.83	93.44	93.47
400	93.7	96.25	94.56	94.58
500	94.53	96.52	94.78	94.56
600	94.89	96.76	94.83	94.84
700	95.76	97.59	95.34	95.45
800	**98.78**	**97.45**	**98.67**	**98.88**

4.11 Comparison of Performance

To compare the performance of our hyper parameter tuning CNN model with other's model based on Precision, Recall, F-Measure value and accuracy is shown in table 6.

Table 6 Comparison of Performance

Model Name	Precision	Recall	F1Score	Accuracy
CNN	84%	84%	89%	88%
Hyper parameter tuning-CNN	**98%**	**97%**	**98%**	**98%**
VGG16	85%	95%	95%	95%
VGG19	94%	94%	94%	94%
RESNET50	88%	88%	87%	88%
RESNET101	89%	88%	88%	88%
RESNET50	89%	88%	88%	88%

To assess the efficiency of our model, we used a variety of

hyper-parameters. We show the graphical representation of our hyper parameter tuning model, which includes model accuracy, model loss, and a classification report on class (1-36).

The model accuracy graph shows that there is a significant difference in train and test accuracy throughout epochs. When comparing the test phase to the high phase of the train, we seem that there is a difference in accuracy between the train and test phases.

The model loss of alphabet class (1-36) is closely connected to the model accuracy of alphabet lass (1- 36), as shown in figure 2.

We show the classification report each alphabet class in Table 7. This shows a deeper representation of the classifier in our model that our train classifier model can identify the test dataset which class has strongly recognized or not.

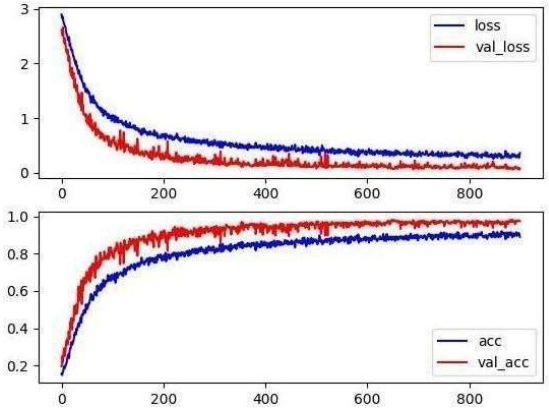

Figure 2 Model Accuracy and Model Loss during trained model

Table 7 Classification report for Hyper parameter tuning-CNN model

	precision	recall	f1-score
ং	0.95	1.00	0.98
ঃ	1.00	1.00	1.00
অ	1.00	1.00	1.00
আ = া	0.95	0.90	0.92
ই = ি	1.00	1.00	1.00
উ = ু	0.95	1.00	0.98
ঊ = ূ	1.00	1.00	1.00
ঋ = ৃ	1.00	1.00	1.00
ঌ	1.00	1.00	1.00
ঐ	0.95	1.00	0.98
ঔ	1.00	1.00	1.00
ব	1.00	0.95	0.97
ভ	1.00	1.00	1.00
ম	1.00	1.00	1.00
য	0.95	1.00	0.98
র	1.00	1.00	1.00
ল	0.95	1.00	0.98
শ	1.00	0.95	0.97
ষ	1.00	0.90	0.95
স	1.00	0.95	0.97
হ	1.00	0.95	0.97
ড়	1.00	1.00	1.00
ঢ়	0.91	1.00	0.95
য়	1.00	0.95	0.97
ৎ	0.91	1.00	0.95
ং	1.00	1.00	1.00
ঃ	1.00	0.95	0.97
ঁ	1.00	0.95	0.97
০	1.00	1.00	1.00
১	1.00	1.00	1.00
২	1.00	1.00	1.00
৩	1.00	1.00	1.00
৪	1.00	0.90	0.95
৫	0.87	1.00	0.93
৬	1.00	1.00	1.00
৯	0.90	0.90	0.90
accuracy			0.98
macro avg	0.98	0.98	0.98
weighted avg	0.98	0.98	0.98

5. CONCLUSION

Approximately 15% of the population has greatly suffered from disability problems in Bangladesh where community of the deaf and dumb would be an asset. Many organizations have already put in time for facilitating the communication with a disable people. Because of insufficient scope this community cannot express their thoughts and feelings with the normal community. Our goal is to develop a method that can accurately recognize the alphabet of Bengali Sign Language. Initially, we have collected a dataset for Bengali sign language which is called Ishara-Lipi Bengali dataset. Then we have augmented this dataset for better improvement. After that, we developed a Convolutional Neural Network with 9layers, dropout layers, and a dense layer. We have experiment our model with various type of hyper parameter tuning. Our model has given a well result that has overall accuracy as 98 percent on the test dataset. Our model has only a 2 percent or 3 percent error rate on each gesture class. In future, we will try to work on both Bengali and English sign language dataset and apply transfer learning technique.

6. REFERENCES

[1] Ahmed, Tauhid S, Akhand M.:Bangladeshi sign language recognition using fingertip position. In: 2016 International Conference on Medical Engineering, Health Informatics and Technology (MediTec). IEEE, pp. 1–5 (2016).

[2] Chen, Lingchen et al.:A survey on hand gesture recognition. In: 2013 International conference on computer sciences and applications. IEEE, pp. 313–316 (2013).

[3] Haque, Promila, Dasb B, Kaspy MN.: Two- Handed Bangla Sign Language Recognition Using Principal Component Analysis (PCA) And KNN Algorithm. In: 2019 International Conference on Electrical, Computer and Communication Engineering (ECCE). IEEE, pp.1–4(2019).

[4] Hoque, Tazimul.:Automated Bangla sign language translation system: Prospects, limitations and applications. In: 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV). IEEE, pp. 856- 862 (2016).

[5] Wang, Chen, Xi Y.: Convolutional neural network for image classification. In: Johns Hopkins University Baltimore, MD 21218 (1997).

[6] Paulo T, Ribeiro F, Reis LP .:Vision-based Portuguese sign language recognition system. In: New Perspectives in In-formation Systems and Technologies, Volume 1. Springer, pp. 605– 617(2014).

[7] Itkarkar, RR and Anil V Nandi.: Hand gesture to speech conversion using Matlab. In: 2013 Fourth International Conference on Computing, Communications and Networking Technologies (ICCCNT). IEEE, pp. 1–4 (2013).

[8] Khan, Asifullah A survey of the recent architectures of deep convolutional neural networks. In: Artificial Intelligence Review 53.8, pp. 5455–5516 (2020).

[9] Huong, Thi T N, Huu T V, Xuan TL.: Static hand gesture recognition for vietnamese sign language (VSL) using principle components analysis". In: 2015 International Conference on Communications, Management and Telecommunications (ComManTel). IEEE, pp.138-

141(2015).

[10] Shinde, Sonajirao S, Rm Aute.:Real Time Hand Gesture Recognition and Voice Conversion System for Deaf and Dumb Per son Based on Image Processing. In: Journal NX 2.9, pp. 39–43 Smith, J. M. and A. B. Jones (2012). Book Title. 7th. Publisher.

[11] Dabre, Kanchan and Dholay S.: Machine learning model for sign language interpretation using webcam images. In: 2014 International Conference on Circuits, Systems, Communication and Information Technolongy Applications (CSCITA). IEEE, pp. 317–321 (2014).

[12] Saha, Nath H.: A machine learning based approach for hand gesture recognition using distinctive feature extraction. In: 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC). IEEE, pp. 91–98 (2018).

[13] Kishore, PVV.: Optical flow hand tracking and active contour hand shape features for continuous sign language recognition with artificial neural networks. In: 2016 IEEE 6th international conference on advanced computing (IACC). IEEE, pp. 346–351(2016).

[14] Prakash, Meena R .:Gesture recognition and finger tip detection for human computer interaction. In: 2017 International Conference on Inno- vations in Information, Embedded and Communication Systems (ICIIECS). IEEE, pp. 1–4 (2017).

[15] Cao, Ming C L, and Yin Z .:American sign language alphabet recognition using microsoft kinect. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops, pp. 44- 52(2015).

[16] Perez-Sanz, Fernando, Pedro J N, and Marcos.: Plant phenomics: an overview of image acquisition technologies and image data analysis algorithms. In: GigaScience 6.11gix092.Smith, J.M. and A. B. Jones (2012). Book Title. 7th. Publisher (2017).

[17] Hussain, Zeshan.: Differential data augmentation techniques for medical imaging classification tasks. In: AMIA Annual Symposium Proceedings. Vol. 2017. American Medical Informatics Association, p. 979 (2017).

[18] Vogelsang, David C and Bradley J.: Erickson Magician's corner: 6. TensorFlow and TensorBoard(2020).

[19] Albawi, Saad, Mohammed TA, and Al-Zawi S.:Understanding of a convolutional neural network". In: 2017 International Conference onEngineering and Technology (ICET). Ieee, pp. 1–6 (2017).

[20] Xu, Xiaowei.:. A deep learning system to screen novel coronavirus disease 2019 pneumonia. In: Engineering 6.10, pp. 1122–1129 (2020).

[21] Alani, Ali A.: Hand gesture recognition using an adapted convol tional neural network with data augmentation. In: 2018 4th International conference on information management (ICIM). IEEE, pp. 5–12 (2018).

[22] Hussain, Soeb.:.Hand gesture recognition using deep learning. In: 2017 International SoC Design Conference (ISOCC). IEEE, pp. 48–49.Jones, A.B. and J. M. Smith (Mar. 2013). Article Title. In: Journal title 13.52, pp. 123–456 (2017).

[23] Zhang, Shanwen.: Cucumber leaf disease identification with global pooling dilated convolutional neural network. In: Computers and Electronics in Agriculture 162, pp. 422–430 (2019).

[24] Hahn, Sangchul, Choi H Understanding dropout as an optimization trick. In: Neurocomputing 398, pp. 64–70 2020).

[25] Xu, Qi.:Overfitting remedy by sparsifying regularization on fully connected layers of CNNs. In: Neurocomputing 328, pp. 69–74 (2019).

[26] Wang, Yulong, Zhang H, and Zhang G PSO- CNN: An efficient PSO-based algorithm for fine- tuning hyper-parameters of convolutional neural networks". In: Swarm and Evolutionary Computation 49, pp. 114–123 (2019).

[27] Bibaeva, V., 2018, September. Using metaheuristics for hyper-parameter optimization of convolutional neural networks. In 2018 IEEE 28Th international workshop on machine learning for signal processing (MLSP) (pp. 1-6). IEEE.

[28] Kligvasser, I., Shaham, T.R. and Michaeli, T., 2018. xunit: Learning a spatial activation function for efficient image restoration. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 2433-2442).

[29] Adithya, V. and Rajesh, R., 2020. A deep convolutional neural network approach for static hand gesture recognition. Procedia computer science, 171, pp.2353-2361.

[30] Ozcan, T. and Basturk, A., 2019. Transfer learning-based convolutional neural networks with heuristic optimization for hand gesture recognition. Neural Computing and Applications, 31,pp.8955-8970.Sannella, M. J. 1994 Constraint Satisfaction and Debugging for Interactive User Interfaces. Doctoral Thesis. UMI Order Number: UMI Order No. GAX95-09398., University of Washington.