Two Dimensional Convolutional Neural Network CNN Approach For Detection of Bangla Sign Language

Pollock Nag
Dept. of Computer Science and Eng.
BRAC University
Dhaka, Bangladesh
pollock.nag@g.bracu.ac.bd

***Abstract*—Sign language is known as the primary communication medium for deaf and mute people. But the lack of available resources and a steep learning curve deter the average person from learning it making communication with the mute and deaf difficult. This problem creates an opportune place for the application of machine learning which has given rise to our emerging field. A large number of papers with high accuracy have already been published for English, French, and other languages. But the number of papers on its application for Bangla Sign language is few. Most of the researchers use SVM, ANN or KNN as classifiers. We chose CNN because it is excellent at high accuracy image classification. In this paper we use a large dataset consisting of 30 classes with 5000 images each totalling to about 15000 images of bangla sign alphabets. Previous works were done only on 10 classes. We began work on those 10 bangla alphabets and later increased the number of classes to 30. We tested the accuracy's of pre trained CNN models such as Dansenet201,VGG16, InceptionV3, Resnet50, MobileNetV2, InceptionResnet, EfficientnetB2 along with our custom CNN model and were able to achieve 97.97%, 96%, 96.22%, 56.44%, 90%, 94%, 4%,98.3 % train accuracy and 86.43%, 88%, 88.33%, 54.50%, 60%, 53%, 4.2%,87% validation accuracy respectively. Our custom cnn model has consistently given better training and validation accuracy than any pre-trained model with lesser layers which in turn require less computations making for a lighter and faster model while maintaing high accuracy.**

Keywords: Bangla sign language, CNN, KNN, ANN, VGG16, Resnet50, InceptionV3.

I. Introduction

Language constitutes a fundamental building block of society. If people could not communicate they would not be able to coordinate and that would hamper positive growth. The ability to talk is such a commonplace concept that we forget that there are people in the world who cannot communicate vocally whether by disability or accident. Sign language stands as the primary standardized method of communication to many such individuals. Sign language is usually conveyed via hand gestures where one or both hands form a specific shape which represents one of the letters in a particular language. Bangladesh has more than 30 lakh people who are hearing impaired [6]. The Center for Disability and Development or CDD recognized Bangla Sign Language as a standard of communication for the Bangla Deaf Community in 2000 yet the facility and opportunities for learning sign language remains inadequate [6]. In recent years research has been conducted to create a machine learning model for recognizing sign language and translating into letters that the general public can understand. With the help of image recognition we can help close the impairment gap by translating sign languages into both written and spoken forms easing communication. Image recognition is performed by a CNN model trained upon a fixed dataset consisting of images of hand signs and their corresponding labels which are the Bangla alphabet. Language is the medium of communication. But in our society some people unfortunately do not have the ability of speaking and listening. To communicate with those people we need to use sign language. In Sign language we generally use various gestures like hand gestures or symbolic gestures instead of sound. As sign language is quite hard for common people to understand, many people lose their interest in learning sign language.

II. Problem statement

In this modern world, computer vision is helping in every sector to make our life easier. In order to develop with this modern world, Deaf people frequently use sign language to communicate . For every language in the world there exists its signed counterpart. Such as, Australian Sign Language (AUSLAN),British Sign Language (BSL), etc. In our country deaf people's language is known as Bangladesh Sign language (BDSL) which is shown in figure 1.1. Sign language is not commonly known making communication with the disabled difficult.This kind of innovation will benefit the deaf people of our country. Fundamental goal of this kind of work is to work as a digital helping device between hearing people and deaf people.

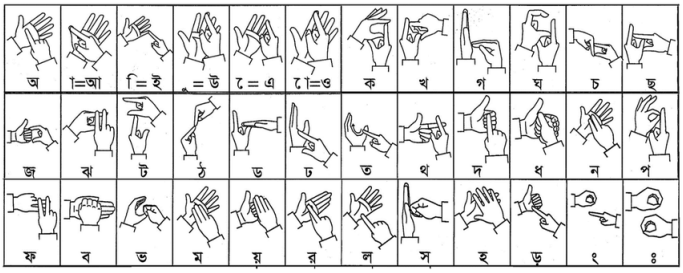

Fig. 1. Sample Data form Dataset

III. Related works

In a project, Sohalia Rahman, Naureen Fatema, M. Rokonuzzaman et al.,[1] use a dotted glove to identify the gesture of hands. The system collects the dots and matches with the pre-prepared charts. The researcher uses image processing on the input image. By this way, they identify numerical letters of BDSL. Using gloves is not an effective method.

Oishee Bintey Hoque, Mohammad Imrul jubair, Md. Saiful Islam, Al-Farabi Akash, Alvin sachine Paulson et al., [2] proposed a paper using faster (R-CNN) which generally makes maps and a network regional proposal network (RPN). This method processes the input picture with a high possibility of containing the desired hand gesture. After this stage, ROI pooling reduces the maps into the identical shape. After dividing the feature map input into a set numeral of roughly equivalent areas it applies max–pooling into each zone. By this way, they said the accuracy of their project

Md. Sanzidul Islam, Sadia Sultana Sharmin Mousumi, Nazmul A. Jessan, AKM Shahariar Azad Rabby, Sayed Akhter Hossain brought in their paper named "Ishara-Lipi: The First Complete Multipurpose Open Access Dataset of Isolated Characters for Bangla Sign Language" [3] that by using ADAM optimizer they got a rate 0.001. Their model of CNN contains 9-layers. For their sign character database, they kept the data for testing 15% and for training 85%. They assert that after 50 epochs, their accuracy on training set and validation was 92.65% and 94.74%.

Md. Islam et al., [4] suggested a paper using Convolutional Neural Network (CNN) to recognize BdSL digits. Firstly they converted the images in jpg format in 128 by 128 pixels for the dataset.Then the dataset was resized by 28 into 28 pixels converted into gray level pixels. Then converted into binary colored pictures given the labels. The model achieved 95.5% training accuracy 94.88% validation accuracy.

Shirin Sultana Shanta, Saif Taifur Anwar et al.,[5] proposed a paper using Convolutional Neural Network (CNN) and SIFT to recognize BdSL. They implement skin masking technique to crop only region of interest(ROI).Then extract feature descriptor using SIFT (Scale Invariant Feature Transform), Use k-means clustering to obtain features as clustered descriptor, use bag of features to represent the features in histogram of visual vocabulary , Input the data in CNN as histogram and check output accuracy is 98.20%.

Md. Islam et al.[6] suggested a paper using CNN in order to recognize Bangla Sign Language.Firstly, they convert the image into a grayscale image then they normalize those images. For normalizing the images they divide gray pixels by max gray level value which is 255. Then, images were reshaped in 64 by 64 pixels for exploration. Finally, they input these images in CNN algorithm where the number of convolution, pooling and fully connected layers are six, three and two (one for input and one for output) accordingly. For basic characters, numerals, and their combined use, they were able to attain accuracy of 99.83 percent, 100 percent, and 99.80 percent, respectively.

Consisting of 24168 samples, the authors Hossain et al., [7] state in their proposed model that they got the highest accuracy from digit detection by putting some extra layers such as convolution layers selected number, max pooling, dropout etc. they also claimed that with 30FPS, their model can give better performance. Moreover, by adding extra layers they solved their detection problem of having a kind of similar input.

To recognize Bangla sign language Lutfun nahar, Nanziba Basnin, and MD Shahadat Hossain et al.,[8] makes use of CNN with LSTM. The background subtraction creates a foreground mask. Grayscale conversion ensures that only a single channel is used, speeding up the learning process. Morphological erosion removes the noise. The image is then run through a median filter and resized. This image is sent to the CNN resulting in a testing accuracy of 88.5%.

F. M. Javed Mehedi Shamrat et al., [9] proposed a paper using Convolutional Neural Network (CNN) and SIFT to recognize BdSL. The system applies transformation on image then applies logarithmic replace technique to control extra light. Every pixel measurement is replaced with the logarithmic grade. Then the LBP is applied to the image.

The paper named Bangla sign Language Recognition using hand Gestures: A Deep Learning Approach by T.B. Das and M.J. Islam et al., [10] proposes a CNN variant-1 with 6 layers. The images are taken with a webcam where the background is segmented through flipping, grayscaling and blurring it without the hand first. Then the hand is introduced which is grayscale, blurred and using a threshold is separated from the image with bitwise AND operation. This results in high accuracy from the model.

IV. Methodology

A. Work Plan

In total of 36 characters are available in Bangla sign language. So, we will collect all the 36 datasets and rearrange them properly. To avoid the chance of desultory, the data sets will be categorized in different folders. Also, the folders will be labeled according to numeric naming convention. For making the datasets usable to machine learning, we will resize the height and width of the images and convert the images to grayscale. We will check each and every datasets and resize them by 128 * 128 pixels. Then, we will try to split the datasets for training and testing purposes. We plan to train our datasets with multiple layers of CNN [**?**] to produce the most accurate results.

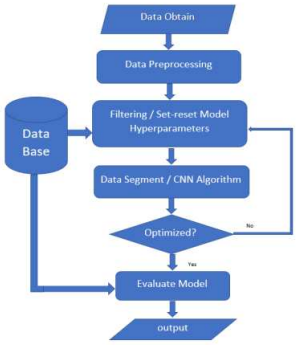

Fig. 2. Working Plan Flowchart

B. Dataset, Libraries & Tools

There are total 30 classes for one-hand representation of Bangla Sign Language. In 30 classes there are total approximately 15000 which is split into 80 percent training and 20 percent test purpose.

C. Architecture

A convolutional neural network (CNN, or ConvNet) is a type of artificial neural network that is extensively used in deep learning to interpret visual information. A Transfer Learning technique is suggested in this work. The size and characteristics of the data set make it ideal for using a transfer learning strategy, allowing a pre-trained CNN with all of its weights to be used to construct a new transfer learning model specialized in detecting sign language with high accuracy. Some of used models are - ***VGG16:*** VGG16 has 16-layer deep neural network (VGGnet). with 138 million parameters. The network's appeal comes from its simplicity. It has a uniform architecture. We can use around 64 alternatives to the number of filters, where we can extend to approximately 128 and then to 256. We can utilize 512 filters in the last phases.

ResNet50: Resnet50 is a 50-layer deep convolutional neural network (CNN) consisting of 48 layers of convolution, 1 MaxPool and 1 Average Pool layer. A residual Neural Network(ResNet) is a sort of Artificial Neural Network (ANN) which constructs a network by pilling up residual blocks on top of one another. In this method, a network is pre-trained on over a million photographs and is stored in the imageNet database. A 224x224x3 image is used as the input, followed by a MaxPooling layer with a 3x3 filter.

Inception-v3: Inception-v3 is design from the inception family that performs factorized 7*7 convolutions, label smoothing, and includes an additional classifier to send label information farther down the network. InceptionV3 model is the outcome of several concepts that different scholars have refined over time. The model has a variety of elements, including symmetric and asymmetric buildings, max pooling, concatenations, dropouts, and completely connected layers. This model heavily relies on batch normalization, and Softmax is utilized to compute loss.

Inception-ResNet-v2: ResNet and Inception provide the best performance in image classification in relation to the computational cost necessary. Inception-ResNet as the name implies combines the Inception and Residual architectures to obtain the best of both worlds.

DenseNet: By concatenating the output feature maps of the layer with the input feature maps rather than calculating their total, DenseNets streamline the connection pattern between layers introduced in previous designs. Due to the lack of duplicate feature maps, DenseNets may operate with fewer parameters than a comparable classical CNN.

D. Data Preprocessing

We make use of the KerasImage Preprocessing library to augment our data.

Firstly, All the input images are resized from their original sizes to a matrix of size 224 by 224. The input dataset contains images with a variety of sizes which cannot be passed through our model due to size mismatch. Resizing allows us to control the size of each layer of our model while allowing the flexibility of having a dataset consist of multiple image sizes. A size of 224 by 224 is a good middle ground between the processing speed and model accuracy.

Secondly, The resulting images are then flattened and converted to grayscale. This reduces the number of channels that need to be processed from 3 to 1 resulting in a decrease in time required for each step from 24s to 8s each and the total time for each epoch from 600s to 200s roughly which is a 3x increase in processing speed from running the same data in rgb.

E. Convolutional Operation

The main factor of CNN is the convolution layer. The network's computational capacity is primarily its responsibility. This layer includes two matrices: the limited

portion of the receptive field and a group of trainable parameters called kernel, and another part is the confined segment of the receptive field. The kernel is less in size than a picture, but it has more depth. This means that if the image contains 3 (RGB) channels, then the height and the width of the kernel are small, but the depth is large. The kernel shifts the height and width of the image throughout a forward run, providing a visual description of the accessible area. At each spatial place in the image, a two-dimensional representation of the kernel's response generates an activation map.A stride refers to the kernel's sliding size. Assuming, our input size is W x W x D and,
F= with a spatial dimension a Dout number of kernels,
S= stride
P= padding amount
Now, to calculate the size of the output volume the formula is:

$$Wout = \frac{W - F + 2P}{S} + 1 \tag{1}$$

The pooling layer uses a summary neighboring outputs to replace the network's output at specific spots. This reduces the size and in turn the amount of calculation required. During the pooling process, each slice of the representation is treated separately. The rectangle neighborhood L2 norm, rectangular neighborhood average and based on distance from the central pixel a sample mean are examples of pooling functions. The most prevalent method, max pooling, reports the neighborhood's maximum output.

The following formula can be used to compute the output volume:

$$Wout = W - FS + 1 \tag{2}$$

Here, The activation map's dimensions in this case are W x W x D, where F is a spatially-sized pooling kernel and S is stride. Pooling provides some uniform translation in all cases, for instance an object may be identified no matter in which it occurs on the panel.

F. Activation Function

Sigmoid: The mathematical version of sigmoid nonlinearity is sigmoid(x) = 1/ (1+e-x). A real-valued number is "squashed" into a value between 0 and 1.
ReLU: In any CNN architecture, activation functions play a critical role in determining which node should be fired. The function ReLU can be represented mathematically as ReLU(x) = max (0, x) (1), where x is the input to a neuron
Softmax: The Softmax Activation Function is a fascinating activation function that takes real-number vectors as input and normalizes them into a probability distribution proportional to the exponentials of the numbers.

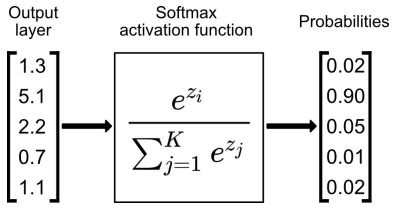

Fig. 3. Convolution Operation

V. Implementation

A. Analysis

Dropout and Batch Normalization: Dropout layers were a common recommendation for reducing over fitting. So we decided to try creating a model with drop out. Batch normalization would ensure our values kept within a certain range. In this model we use 5 layers. (2 Convolution layers with max pooling, 3 dense layers). All layers include batch normalization with axis= -1.All layers include a Dropout layer of 0.2.
Simpler CNN with 7 layers: It was apparent from testing the pre-trained models that we would require a simple model for our dataset. So one of the cnns we tried was one with 7 very simple layers. In this model we have 7 layers which are 2 Convolution layers, 2 max-polling layers, 2 batch-normalization layers, and 1 dropout layer. However, we have tried few others configurations as well. Among them configuration 2 gives us better accuracy. So the model summary of configuration 2 is attaching below
These are the Train and Test score of our models -
In **Table I**, the Train and Test accuracy and loss graph for each model are given.

Model	Top Training Accuracy	Top Validation Accuracy
Custom CNN 7Layers with Normalization	98.3%	87%
DenseNet201	97.97%	86.43%
InceptionV3	96.22%	88.33%
ResNet50	56.44%	54.50%
VGG16	96%	88%
MobileNetV2	90%	60%
InceptionResnet	94%	53%
EfficientNetB2	4%	4.2%

TABLE I
MODEL ACCURACY AND LOSS

B. Result

we know that a loss function graph can be used to determine if a model is over-fitting or not. When models rely too much on their training data and lose their capacity to perform effectively on new data, this is known as over-fitting. In other words, the model includes and learns from the random oscillations in the training data. We might have had minor over-fitting concerns, as shown by the loss function graphs of the classic transfer learning models. These over-fitting problems had an effect on the predictability of outcomes. This problem was resolved via normalizing and the addition of few dropout layers, which increased the prediction accuracy of the Bangla Sign Language categorization. Additionally, the total number of layers in our unique CNN model is 7, which makes it lighter and faster

to train. Finally, as seen in figure 16 and figure 17 , our customized CNN model performs better than any other pre-trained models and provides us with greater accuracy.

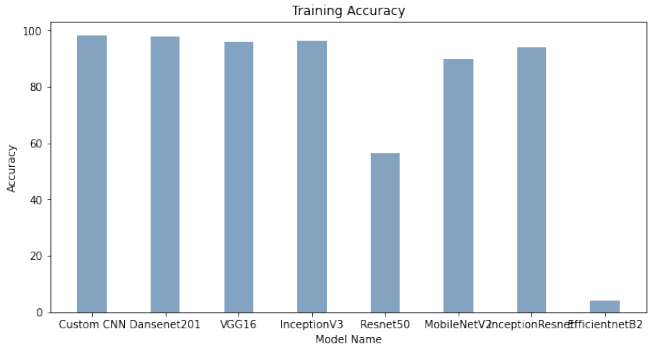

Fig. 4. Training Accuracyl

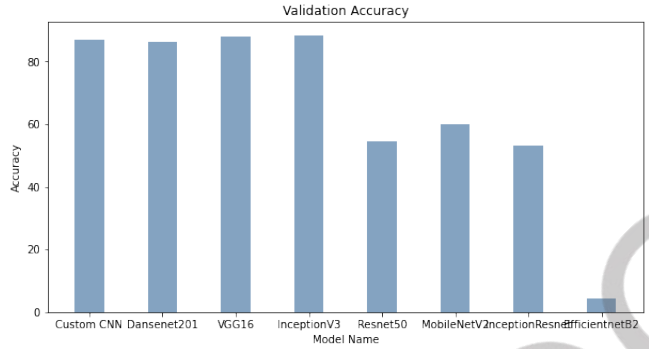

Fig. 5. Validation Accuracy

C. Loss Function Graph

A loss function is a metric that measures how well a prediction model predicts the predicted outcome or value. The learning issue is transformed into an optimization problem, a loss function is defined, and the method is optimized to minimize the loss function.

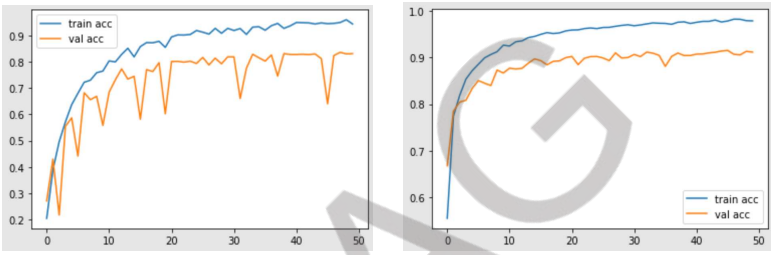

Fig. 6. Custom CNN Accuracy DanseNet Accuracy

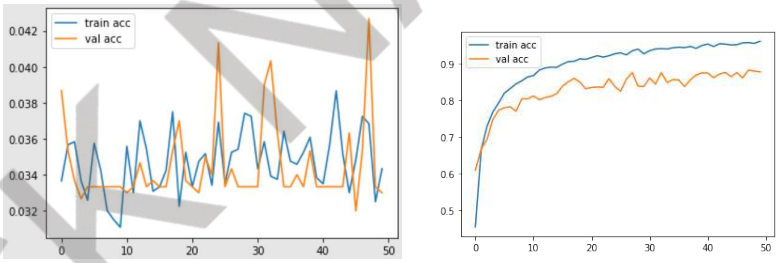

Fig. 7. Efficient Accuracy Inception Accuracy

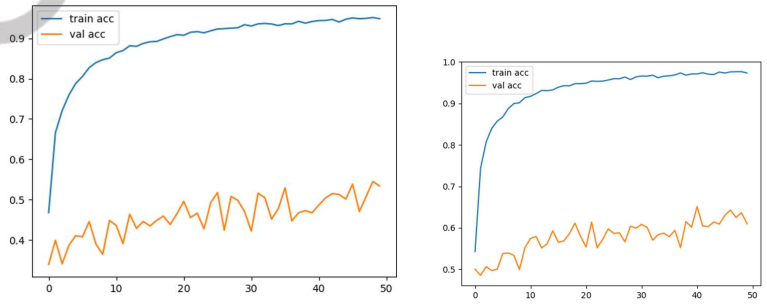

Fig. 8. InceptiopnResNet Accuracy MobileNet Accuracy

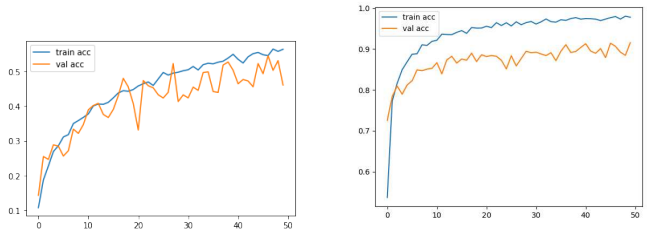

Fig. 9. ResNet Accuracy VGG-16 Accuracy

VI. Conclusion

A. Discussion

we worked on 15000 images of hand gestures. All these images were augmented to increase the number of datasets. Then the dataset was split into two folders using split-folders library. One is train dataset which contains 80 percent of the image of whole dataset. The validation dataset which contains rest 20 percent images. Then we perform various pre-trained model on this dataset. We use Dansenet201, VGG16, InceptionV3,Resnet50, MobileNetV2, InceptionResnet, EfficientnetB2. Among them DenseNet201 gives the highest training and validation accuracy 98 percent and 86 percent accordingly. To satisfy our goal we tried to build a custom CNN model. Which includes in total 7 layers. To improve the accuracy and reduce overfitting issue, we added several dropout layers and normalizing layers. Then we comes with a model which gives better accuracy (98.3 percent training and 87 percent validation) than any tested pre-trained model ensuring less depth of neurons, lighter model and faster training time.

B. Limitations

Even though we have successfully run the system we got some limitations. Due to tyhe shortage of processing resources, for instance, the GPU's limits considerably took longer training phase time to run. Moreover, the employed dataset in our project was pretty large. Furthermore, We also faced challenges in classification as well as prediction for training lesser datasets. However, We were able to find out this problem on time and improved it by adding new data sets with existing datasets.

C. Future Works

We intend to work with all Bengali alphabets in the future. But we also want to create a web application which can be able to identify bangla sign language in real time. Additionally, the same program will be able to differentiate between different bangla sign languages. We have had over-fitting problems in our current system. We'll take care of this by enhancing the data.

D. Conclusion

Nowadays, sign language is very essential for people who have problems in hearing and talking. This dataset contains Bangla Sign Language which is developed by using convolutional neural networks. We use CNN architecture for faster delivery. This system will be the digital interpreter between deaf and normal people which can be easier for understand the language to deaf and also very friendly for hearing people. Initially, this paper contains Sign language of all the Bangla Alphabets and further we will also work for the Bangla Numerical Signs. Conducting this procedure as friendly as possible for both hearing and non-hearing people is our main objective.

E. Acknowledgement

At first, With the blessings of God, we have accomplished of our thesis paper so far.
Secondly, we are gratified for the genuine support and supervision of our supervisor Dewan Ziaul Karim and co-supervisor Mr. Rafeed Rahman sir, in the field of collecting the literature and data. Moreover, their motivation have made us to state of our model and the successfull completion of our paper.
Finally, With the kind support and prayer of our parents We are on the edge of our graduating. It might not be possible without their support and kind help.

References

[1] S. Rahman, N. Fatema, and M. Rokonuzzaman, "Intelligent assistants for speech impaired people," in ICCIT, 2002.

[2] O. B. Hoque, M. I. Jubair, M. S. Islam, A.-F. Akash, and A. S. Paulson, "Real time bangladeshi sign language detection using faster r-cnn," in 2018 international conference on innovation in engineering and technology (ICIET), IEEE, 2018, pp. 1–6.

[3] M. S. Islam, S. S. S. Mousumi, N. A. Jessan, A. S. A. Rabby, and S. A. Hossain, "Ishara-lipi: The first complete multipurposeopen access dataset of isolated characters for bangla sign language," in 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), IEEE, 2018, pp. 1–4.

[4] S. Islam, S. S. S. Mousumi, A. S. A. Rabby, S. A. Hossain, and S. Abujar, "A potent model to recognize bangla sign language digits using convolutional neural network," Procedia computer science, vol. 143, pp. 611–618, 2018.

[5] S. S. Shanta, S. T. Anwar, and M. R. Kabir, "Bangla sign language detection using sift and cnn," in 2018 9th international conference on computing, communication and networking technologies (ICCCNT), IEEE, 2018, pp. 1–6.

[6] M. S. Islalm, M. M. Rahman, M. H. Rahman, M. Arifuzzaman, R. Sassi, and M. Aktaruzzaman, "Recognition bangla sign language using convolutional neural network," in 2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), 2019, pp. 1–6. doi: 10.1109/3ICT.2019.8910301.

[7] S. Hossain, D. Sarma, T. Mittra, M. N. Alam, I. Saha, and F. T. Johora, "Bengali hand sign gestures recognition using convolutional neural network," in 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), IEEE, 2020, pp. 636–641.

[8] N. Basnin, L. Nahar, and M. S. Hossain, "An integrated cnn-lstm model for bangla lexical sign language recognition," in Proceedings of International Conference on Trends in Computational and Cognitive Engineering, Springer, 2021, pp. 695–707.

[9] F. Shamrat, S. Chakraborty, M. M. Billah, M. Kabir, N. S. Shadin, and S. Sanjana, "Bangla numerical sign language recognition using convolutional neural networks," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 1, pp. 405–413, 2021.

[10] T. Das and M. Islam, "Bangla sign language alphabet recognition using hand gestures: A deep learning approach,"