# Bangla Sign Language Recognition Using Angular Margin

Towsif Alam Chowdhury, Samiya Kabir Youme, and Hossain Ahamed

***Abstract*—Sign language is the only way of communication for the deaf and mute people. Most normal people do not understand sign language as a result the deaf and mute community faces great difficulty to communicate with people. To address this problem different machine learning methods had been used in the past among which deep learning based methods excels the most. For recognition of sign languages like English, French, Japanese, etc. a lot of work has been done so far. However for Bangla Language, which is one of the most spoken languages, much significant work has not been done. One of the main reasons for this is the lack of a large enough standard dataset. In this study we have used a large dataset of 30916 hand images of Bangla characters consisting of both numerals and alphabets. Almost all prior works for Bangla Sign Language used very small datasets which are not ideal for deep learning based methods and all the prior deep learning based approaches used Softmax loss function for classification. In our paper we have implemented the Arcface loss function which has not been used in any work for sign language classification yet and we have also compared the performance of Arcface and Softmax loss function. Our proposed method using Arcface loss function gave 99.14%, 99.58% and 99.38% accuracy for digits, alphabets and combined respectively.**

***Index Terms*—BdSL, Arcface, Neural Networks, Sign Language**

## I. Introduction

Languages provide us the ability to express ourselves. Nevertheless, there are some individuals who are deprived of the ability to speak and hear. People with such disabilities are considered as deaf and mute. These people can’t use normal languages to communicate with people so their only source of communication is a special gesture-based method called sign language for communication. Sign languages are different for every country like our normal languages, different languages have their own unique sign languages. The sign language used for Bangla language consists of 10 gestures for digits, 36 gestures for alphabets, and about 4000 double hand symbols for commonly used Bengal words [1]. Around 3 million hearing-impaired people live in Bangladesh [17]. Deaf and mute people often face difficulty communicating with normal people because most people can not interpret sign languages. Since their disability they couldn’t express their talents, as a result, the deaf and mute community is left behind in various tasks. Though modern society is assisting them in education, culture, communication to build their future. However, on busy days it is difficult to learn sign language for normal people. Therefore to fill up the gap, Sign language recognition is much needed for effortless interaction with the deaf and mute community. We address this problem effectively with the help of deep learning as deep learning in some cases can outperform human beings.

Image-based deep learning techniques have closed the gap between the deaf and mute community and the general people. This technique can be used to interpret hand gestures in text or audio form. However, Deep learning methods require an excessive amount of data to provide remarkable results. For popular languages like English, French, Spanish, etc. there are standard benchmark datasets but for Bangla Sign Language no standard dataset is currently available. We have used a dataset comprised of 30916 samples (basic characters: 23864 and numerals: 7052) which were developed by Islam et al. [15]. Most of the prior work done for Bangla Sign Language detection lacks a large enough dataset. So far, the prior works done for the recognition of Bangla Sign Language using Deep Learning used the Softmax function for the classification of classes. In this study, we used a convolutional network-based architecture and used the Arcface loss function for classification, which performs better for classification of Bangla Sign Language characters as it reduces the intraclass distance and increases the inter-class distance. We have also presented a comparison of ArcFace and Softmax loss function using our model.

## II. Literature Review

Along with roughly 6,500 languages, sign language is also considered as another complex language. From the 20thcentury, researchers and developers of deep learning become highly interested in recognition of sign language, as a result many systems have been designed with different languages which includes American Sign Language, Indian Sign Language, Bhutanese Sign Language, Japanese Sign Language, Arabic Sign Language, Australian Sign Language, Srilankan Sign Language and many more. Numerous works have been done due to availability of dataset in these domains mostly for American Sign Language. Previously, glove based methods were used for data collections in which parameters were a bit different for instance, hand’s position, location, angle etc. and high level of processings were needed which made the system complex; still accuracy doesn’t come up to the mark. However, various researchers used various types of dataset to express hand gestures for sign language recognition like, one-handed bangla sign recognition, two-handed bangla sign recognition, specific bangla sign word recognition etc. Since 1990, lots of work has been conducted using statistical methods, machine learning, deep learning. In this section, we review some different approaches for classification of hand sign language recognition of different languages mainly on Bangla.

Different countries use HMM(Hidden Markov Model) including America for sign language but the difference was in their feature extraction strategies [2, 3, 4, 5]. These papers all discussed dynamic gestures. Most used classifiers for BdSL are Machine learning based such as SVM (Support Vector Machine), ANN (Artificial Neural Network) or KNN (K-Nearest Neighbors). Rekha et al. suggested KNN, SVM approach for ISL (Indian sign Language) using two handed sign language dataset [5]. In this [5] paper, they also extended their work with DTW(Dynamic Time Warping) for dynamic gestures. Nikhat et al. proposed ANN to classify five Bangla Hand gesture signs[5]. Chowdhury et al. converted the BdSL to Text using ANN and SVM [7]. Atfirst, they extracted the feature from the input image using contour feature extraction and used SVM for classification. After classification, they used a 5 layer Neural Network to convert letters into text. Deb et al proposed a method with two steps: one is refinement and the other is recognition, using Two Handed Bangla sign language dataset [8]. Firstly, they preprocessed the data by color segmentation of RGB, image Centroid calculation and localized the edge of two wristbands. Then through normalization, recognition of the image is done and after that each hand sign is fitted to equal size for template matching using statistical methods like cross correlation for finding the most appropriate match.

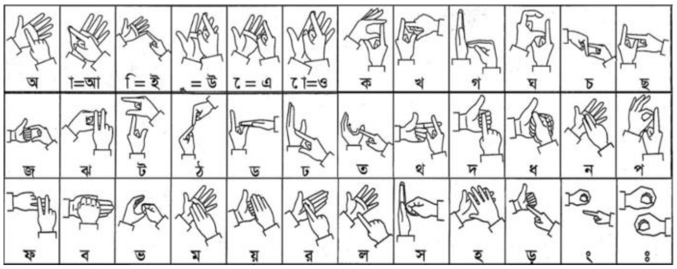

Fig. 1. Bangla Sign Language

The dataset of Ishara-Lipi has 50 sets of 36 different Bangla Sign characters [9]. After preprocessing, they were left with 1800 images resized by 128x128 and the images were turned into grayscale images. Their goal was to create the BdSL (Bangla Sign Language) open access data set so that their work can be used for AI (artificial Intelligence ) based models. They used Otsu's Method to determine the threshold of image automatically and for Feature extraction they used 9 layers CNN. They used Categorical Cross Entropy for loss function so that converging becomes faster to the Global minimum. In this paper [10] the author Rony et.al gave another solution for Bangla sign language and divided the solution in 4 modules first one is Image Data collection for training for this they used tensorflow Inception V-3 model of GoogleNet [google net]. Then they created a classification model using CNN. Third module is all about capturing the real time image from a webcam. For this they used OpenCV and the last module is to test their created CNN model[10].

Sultana et al. are the first user of CNN classifier for BdSL [11]. They presented that by using preprocessing techniques such as Skin masking Edge Detection and SIFT feature extraction, K-Mean clustering, they could obtain the visual bag of words and then fed into the CNN as input as a result, performance of the CNN classifier improved. Rahman et al. used image processing techniques with Feed forward neural networks to detect BdSL sign Language [12]. Yasir et al solved two handed hand sign Recognition for BdSL using PCA (Principal Component Analysis), and LDA(Linear Discriminant Analysis) for feature extraction and also ANN for hand sign classification [13]. In the above mentioned papers, they used normal image processing techniques to extract features before feeding to the neural network [11,12,13]. Potential CNN model is proposed by Sanzidul et al by using the dataset of Ishara-Lipi and they classified the 10 digit of BdSL by operating two subLayers which provide novelty of their work [9]. Aysha et al. proposed a Fuzzy rule based model to classify only 2 types of letters [14]. Islam et al. proposed CNN with batch normalization after Convolution layer and used softmax to reach accuracy 99.80% [15].

In the above discussion, every work was based on different models for instance, Machine Learning based algorithms, Neural networks, CNN and HMM. They all focused on the models mainly. However, we attempted to optimize CNN architecture by operating different loss functions for this research. In our proposed model, we have modified some architectural layers in VGG_16 and then we used angular loss for our model i.e. Arc Face loss [16]. Though angular loss is usually used for FR (Facial Recognition), we attempted to merge ArcFace with our proposed model. We will discuss the proposed Bengali Sign Language recognition model in detail in the methodology section.

## III. Methodology

Sign language recognition can be solved by different algorithms, however, as it is a pattern recognition problem we attempted to resolve by using CNN. The fundamental step needed in our work is outlined in figure 4. In this section dataset properties, our proposed method using Loss function and training details is discussed elaborately.

### *A. Dataset*

For our model We have used a dataset Islam et al [15]. They developed this BdSL dataset of a total of 30916 samples from which 23864 samples are for basic bangla character and 7052 samples are for digits. This dataset has been collected from 25 male students and their age is around (18-26). It consists of two handed images and it is already preprocessed i.e. dataset is normalized and the size for each image is 100×100 pixel and the channel is 3 as this is an RGB image. Some Image samples are shown in the figure y. There are 45 folders of images, we labeled them by numbers (0, 1, 2, 3. . . 44) as sequentially presented as Bangla sign digits and characters sequence. All images are kept in .jpg format. Here, one of the bangla letters is missing i.e. . So 35 Bangla alphabet and 10 bangla digit samples are trained.

### *B. Proposed Model*

As we need to classify multi-class images, CNN based models perform the best in this case. Therefore, our Proposed model includes CNN based Architecture. In figure x represents our architecture for Classifying BdSL dataset.

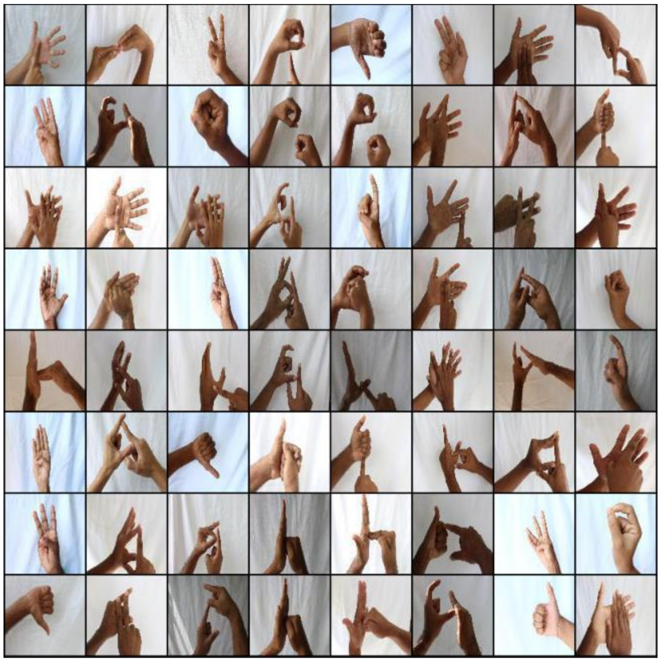

Fig. 2. Dataset for the model provided by Islam et al [15]

1) *Convolutional Neural Network:* Convolutional neural networks are very capable of identifying patterns in the input images that include lines, gradients, circles, or even eyes and faces. For computer vision, it is this property that makes convolutional neural networks so strong. A convolutional neural network is a special form of feed-forward neural network, however, a special type of layer called the convolutional layer provides the strength of a convolutional neural network. It contains many convolutional layers layered on top of each other, and every layer is capable of recognizing patterns. Convolutional neural networks can act directly on a raw image and do not require any preprocessing, unlike earlier computer vision algorithms.

InputLayer: The preprocessed data is readily fed into the network with the help of input layer. The input layer contains 100×100 pixels so 10000 nodes with a number of channels 3 for RGB image.

Convolutional Layer: With the support of an arbitrary number of learnable kernels that slide along the width and height of the input image, a Convolution layer passes the input image, returning a feature chart. A kernel can be considered to be an array of numbers that are also called weights. The size of the kernel we have chosen is 3×3, keeping padding and stride 1 for every layer. There are 13 Convolution layers in our architecture.

Batch Normalization: Batch normalization is a method for training very deep neural networks that provides a consistent input to a layer for each mini-batch. This has the effect of stabilizing the learning process and significantly reducing the number of training cycles necessary for deep networks to be trained.The batch normalization is followed by each convolution layer, and for this model it works remarkably as it is primarily responsible for reducing internal covariate shift and speeding up the training phase.

Activation Function: In any CNN architecture that is susceptible to sort out which node should be activated, there it plays a major role. We use ReLU activation functions in our CNN []. However, for trial and error we have also tried tanh for non-linearity but it doesn't work well. ReLUs are much quicker in training than their alternatives (sigmoid, tanh) [tanh] and can minimize the problem of the vanishing gradient. Mathematically, it is possible to express the ReLU function as:

$$ReLU(x)=max(0,x) \qquad \text{Eq.1}$$

where, x denotes the input to a neuron.

PoolingLayer: Our network uses max-pooling by reducing the number of ties between convolutional layers, this operation reduces the computational burden present in the input feature vector. It reduces the dimensionality while not affecting the channel. It speeds up the process of training and helps to reduce the amount of memory taken from the network. For the max-pooling approach, we selected the kernel size 2×2 and stride is also 2.

$$Nout = floor\ (Nin - FS)+1 \qquad \text{Eq.2}$$

Where, the dimensions of the input image, filter size, and stride size, respectively, are denoted by Nin , F and S.

Fully Connected Layer: The output of the GAP layer at the end of a CNN serves as the input to the fully connected (FC) layer. Convolution can be considered as the FC layer, a layer with a filter size of 1 by 1. The previous layers (convolution, pooling) carry information about local features such as edges, blobs, shapes, etc. in the input image. Such characteristics (matrix, tensor) are flattened into a vector that it fed into the layer of FC. Every neuron in one layer is connected to every neuron in another layer in the FC layer also called dense layer. It carries out a category based on the characteristics gathered from previous layers. We have also applied two hidden layers for better recognition. We have used Stochastic Gradient Descent (SGD) optimizer for our model.

Output Layer: It is the job of the output layer of a CNN to yield the probability of each class given the input image. We set out our utmost FC layer to retain the same number of neurons as there are groups to achieve these probabilities. Then FC layers neurons are fed into Arc Face loss and after that cross entropy is implemented to get output.

In figure x, there are 13 Convolution layers and each CONV layer has a filter. Here, Input image passes every ConvLayer passes through Batch normalization and ReLU activation function then goes through max pooling. There are 2 Fully connected and each fully connected layer is passed through batch normalization and ReLU activation function. Then the Embedding vector is processed through L2 normalization and passed through the Arc Face loss and Output is produced. More details of the Arc Face loss will be discussed in the ArcFace portion.

*a) 3.2.1 Additive Angular Margin Loss:* For our model we have used the angular Loss function to add more discriminating power among the different classes [arc face]. Though it is usually used in Face detection but we have implemented loss function appropriately for our classification model.

For angular loss function, Weights and Embedding vectors need to be Normalized to 1 so that loss function only depends

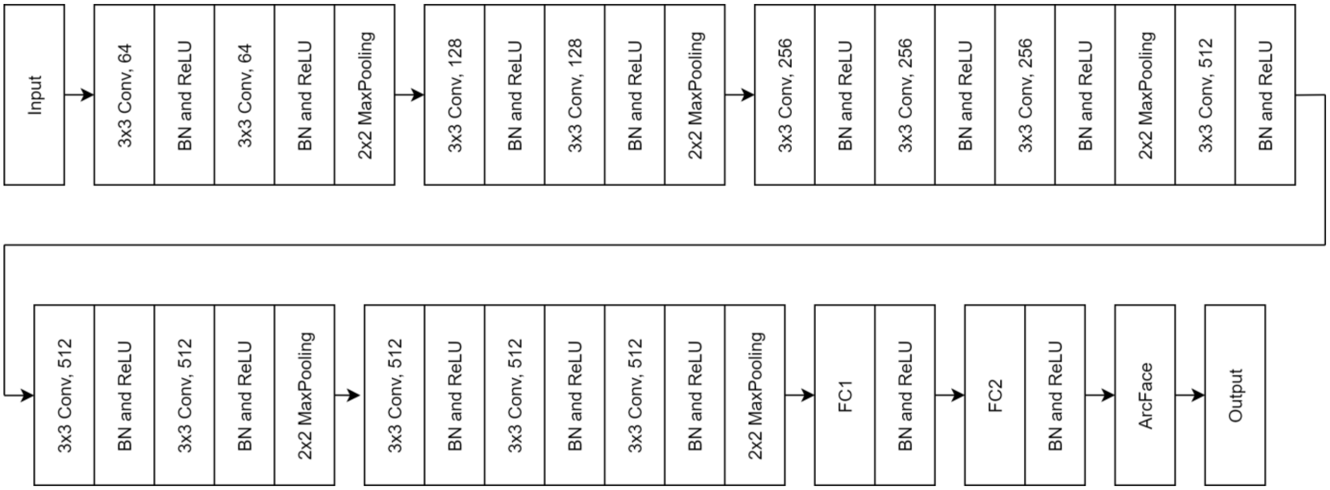

Fig. 3. Our Proposed Model

on the angle between embedding vector and row vector and Bias are also considered as 0 for simplicity in Eq. 1.

$$WjTxi=\|Wj\|\ \|xi\|\cos(\theta) \qquad \text{Eq.3}$$

Then we calculate cos(θ) and after that we add a margin, m so that inter class distance from different classes increases by using Eq. 2. Then we rescale it by multiplying s another constant to project the vectors on s, radius on Hypersphere.

$$\cos(\theta+m) = \cos(\theta)\cos(m)-\sin(\theta)\sin(m) \qquad \text{Eq.4}$$

### C. *Training Details*

We have divided our dataset in three sets randomly: train set, validation set and test set by ratio of 80:10:10. As our class of sign language is 45 which is very small compared to Face recognition so we kept s as 2, and margin is the same as original ArcFace i.e. 0.5. Here we have used ArcFace so to optimize both parameters of ArcFace and model we use SGD with learning rate 0.01 and weight decay 0.01. We didn't use ADAM optimizer because its momentum is quite high. The weight and biases of the model are randomly initialized. We train our model up to 15 epochs with batch size, 100. After trial and error we updated all hyperparameters. The experiments were conducted on a PC with an 8th gen Intel core i5 and NVIDIA 1050Ti 4 G and 8 GB RAM. In every epoch, the inference time of our model was around 1 second. After completion of training of the model, it's performance was evaluated by the remaining fold of dataset that has not been used for training

## IV. Results and discussion

Our dataset contains a total of 30916 images of which 7052 are numeral characters and 23864 are alphabets. We have implemented both Arcface and Softmax loss function on this dataset. Both of the loss functions gave us similar results. The Arcface loss gave us an accuracy of 99.28% on the test set and the Softmax loss gave us 99.74% accuracy on the test data. We have also tested our model separately on the alphabets and the digits. Table II shows the details of our outputs.

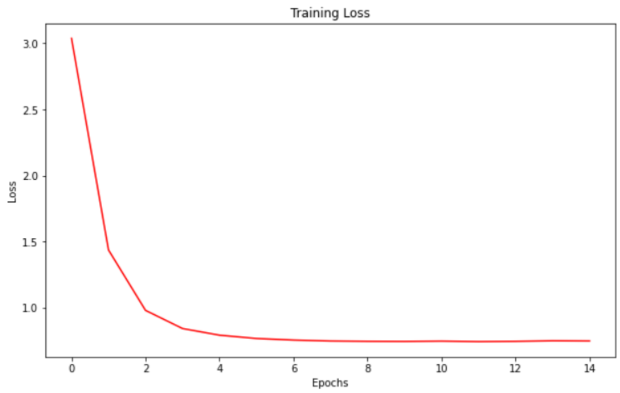

Fig. 4. Loss curve

TABLE I
Proposed model outputs

| Dataset | Training Accuracy(%) | Validation Accuracy(%) | Test Accuracy(%) |
|---|---|---|---|
| Digits | 100.0 | 98.32 | 99.14 |
| Alphabets | 99.92 | 99.46 | 99.58 |
| Combined | 99.85 | 98.96 | 99.38 |

TABLE II
Comparision between Arcface and Softmax loss function

| Dataset | Arcface Test Accuracy(%) | Softmax Test Accuracy(%) |
|---|---|---|
| Digits | 99.14 | 99.43 |
| Alphabets | 99.58 | 99.66 |
| Combined | 99.38 | 99.74 |

TABLE III
COMPARISON BETWEEN EXISTING AND PROPOSED SYSTEMS

| Model | Data set size | Accuracy(%) |
|---|---|---|
| PCA | 2020 | 76.9 |
| LDA | 2000 | 88.55 |
| KNN | 7200 | 96.46 |
| SVM | 4800 | 97.7 |
| Proposed model | 30916 | 99.82 |

Initially, in our model, we didn't use any normalization layers. This gave us inferior performance. The use of normalization layers increased our model performance. We have also used an average pooling layer in between the fully connected layers which we removed later and that increased our model performance by a good margin. Our model performed uniformly well for alphabet data, digit data, and the combined data of both alphabets and digits.

## V. Conclusion

More than 70000 people use Sign Language for their communication, deep learning creates a platform for them by using recognition algorithms to detect hand gestures as for them hand is the source of communication. In this study, convolutional neural networks are utilized to recognize different gestures of hand which is used to represent different bangla letters particularly. For enhancing the discriminative features of hand gestures we implemented ArcFace learned via CNN. In the comprehensive way, we have compared our method with different methods in terms of accuracy and we have also compared with normally used Softmax and Arc Face.

## VI. Reference

1) Bengali Sign Language Dictionary, National Centre for Special Education, Ministry of Social Welfare in cooperation with the Norwegian Association of The Deaf and Bangladesh National Federation of The Deaf, 1997.
2) Starner, Thad & Group, Massachusetts. (1995). Visual Recognition of American Sign Language Using Hidden Markov Models.
3) Kulkarni, Vaishali & Dr, S.D.Lokhande. (2010). Appearance Based Recognition of American Sign Language Using Gesture Segmentation. International Journal on Computer Science and Engineering.
4) Zaki, Mahmoud & Shaheen, Samir. (2011). Sign language recognition using a combination of new vision based features. Pattern Recognition Letters. 32. 572-577. 10.1016/j.patrec.2010.11.013.
5) Chen F-S, Fu C-M, Huang C-L (2003) Hand gesture recognition using a real-time tracking method and hidden Markov models. Image Vis Comput 21:745–758
6) Jayaprakash, Rekha & Majumder, Somajyoti. (2011). Hand Gesture Recognition for Sign Language: A New Hybrid Approach. 1.
7) A. R. Chowdhury, A. Biswas, S. M. F. Hasan, T. M. Rahman and J. Uddin, "Bengali Sign language to text conversion using artificial neural network and support vector machine," 2017 3rd International Conference on Electrical Information and Communication Technology (EICT), Khulna, 2017, pp. 1-4, doi: 10.1109/EICT.2017.8275248.
8) Kaushik Deb, Dr. Muhammad Ibrahim Khan, Helena Parvin Mony, Sujan Chowdhury, D. (2012). Two-Handed Sign Language Recognition for Bangla Character Using Normalized Cross Correlation. Global Journal Of Computer Science And Technology, . Retrieved from https://computerresearch.org/index.php/computer/article/view/446
9) M. Sanzidul Islam, S. Sultana Sharmin Mousumi, N. A. Jessan, A. Shahariar Azad Rabby and S. Akhter Hossain, "Ishara-Lipi: The First Complete MultipurposeOpen Access Dataset of Isolated Characters for Bangla Sign Language," 2018 International Conference on Bangla Speech and Language Processing (ICBSLP), Sylhet, 2018, pp. 1-4, doi: 10.1109/ICBSLP.2018.8554466.
10) A. J. Rony, K. H. Saikat, M. Tanzeem and F. M. Rahat Hasan Robi, "An Effective Approach to Communicate with the Deaf and Mute People by Recognizing Characters of One-hand Bangla Sign Language Using Convolutional Neural-Network," 2018 4th International Conference on Electrical Engineering and Information & Communication Technology (iCEEiCT), Dhaka, Bangladesh, 2018, pp. 74-79, doi: 10.1109/CEEICT.2018.8628158.
11) S. S. Shanta, S. T. Anwar and M. R. Kabir, "Bangla Sign Language Detection Using SIFT and CNN," 2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Bangalore, 2018, pp. 1-6, doi: 10.1109/ICCCNT.2018.8493915.
12) M. A. Rahaman, M. Jasim, M. H. Ali and M. Hasanuzzaman, "Computer vision based Bengali sign words recognition using contour analysis," 2015 18th International Conference on Computer and Information Technology (ICCIT), Dhaka, 2015, pp. 335-340, doi: 10.1109/ICCITechn.2015.7488092.
13) R. Yasir and R. A. Khan, "Two-handed hand gesture recognition forbangla sign language using lda and ann," in Software, Knowledge, Information Management and Applications (SKIMA), 2014 8th InternationalConference on. IEEE, 2014, pp. 1–5
14) Ayshee, Tanzila & Raka, Sadia & Hasib, Quazi & Hossain, Md & Rahman, Mohammad. (2014). Fuzzy rule-based hand gesture recognition for bengali characters. Souvenir of the 2014 IEEE International Advance Computing Conference, IACC 2014. 484-489. 10.1109/IAdCC.2014.6779372.
15) M. S. Islalm, M. M. Rahman, M. H. Rahman, M. Arifuzzaman, R. Sassi and M. Aktaruzzaman, "Recognition Bangla Sign Language using Convolutional Neural Network," 2019 International Conference on Innovation and Intelligence for Informatics, Computing, and Technologies (3ICT), Sakhier, Bahrain, 2019, pp. 1-6, doi: 10.1109/3ICT.2019.8910301.
16) J. Deng, J. Guo, N. Xue and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 2019, pp. 4685-4694, doi: 10.1109/CVPR.2019.00482.
17) Alauddin, Mohammad & Joarder, md abul hasnat. (2004). Deafness in Bangladesh. 10.1007/978-4-431-68397-1_13.