Recognition of Bangladeshi Sign Language (BdSL) Words using Deep Convolutional Neural Networks (DCNNs)

6 authors, including:

Rishad Amin Pulok
Metropolitan University
2 PUBLICATIONS   15 CITATIONS

Md Mizanur Rahman
Daffodil International University
6 PUBLICATIONS   16 CITATIONS

Nusrat Khan
Daffodil International University
6 PUBLICATIONS   20 CITATIONS

Shamsul Haque
Monash University Malaysia
86 PUBLICATIONS   3,225 CITATIONS

Recognition of Bangladeshi Sign Language (BdSL) Words using Deep Convolutional Neural Networks (DCNNs)

Aminul Haque [1], Rishad Amin Pulok [2], Md. Mizanur Rahman [1], Sanzida Akter [1], Nusrat Khan [1], Shamsul Haque [3*]

[1] *Faculty of Science and Information Technology, Daffodil International University, Birulia, Savar, 1216 Dhaka, Bangladesh*

[2] *School of Science and Technology, Metropolitan University, Bateshwar, 3104, Sylhet, Bangladesh.*

[3] *Jeffrey Cheah School of Medicine and Health Sciences, Monash University Malaysia, Jalan Lagoon Selatan, 47500 Bandar Sunway, Selangor D E, Malaysia.*

Abstract

In a world where effective communication is fundamental, individuals who are Deaf and Dumb (D&D) often face unique challenges due to their primary mode of communication—sign language. Despite the interpreters' invaluable roles, their lack of availability causes communication difficulties for the D&D individuals. This study explores whether the field of Human-Computer Interaction (HCI) could be a potential solution. The primary objective is to assist D&D individuals with computer applications that could act as mediators to bridge the communication gap between them and the wider hearing population. To ensure their independent communication, we propose an automated system that could detect specific Bangla Sign Language (BdSL) words, addressing a critical gap in the sign language detection and recognition literature. Our approach leverages deep learning and transfer learning principles to convert webcam-captured hand gestures into textual representations in real-time. The model's development and assessment rest upon 992 images created by the authors, categorized into ten distinct classes representing various BdSL words. Our findings show the DenseNet201 and ResNet50-V2 models achieve promising training and testing accuracies of 99% and 93%, respectively.

1- Introduction

Communication is the essential conduit for transmitting information between individuals [1]. Across the globe, diverse communication systems have emerged, each supporting a distinct mode of expression. Among these, sign language emerged as a visual medium, facilitating communication via complex gestures. This unique form of expression employs an array of hand gestures and physical interactions to convey information. While normal-hearing people engage in interpersonal communication through natural spoken languages, individuals who are Deaf & Dumb (D&D) utilize tactile sign language [2]. Like spoken languages, sign language has its grammatical structures and lexicon, distinguishing it as a distinct linguistic system that is often difficult for hearing people to understand. However, there is a communication gap for D&D people since they need help to speak or hear conventional voice dialogues. D&D individuals distinguish themselves from others with disabilities by demonstrating unique proficiency in tasks where others might struggle. However, their challenges mainly arise in communication with the non-D&D population. D&D individuals effortlessly communicate with other D&D people due to their shared sign language knowledge. However, within the broader social context, they often face communication difficulties, leading to a general sense of doubt [3].

Sign language exhibits extensive diversity [4], varying significantly across nations and cultures. For example, American Sign Language (ASL), British Sign Language (BSL), Bangladeshi Sign Language (BdSL), Japanese Sign Language, French Sign Language, and many more [5]. While these languages share certain similarities, they also possess distinctive features. Consequently, comprehending and distinguishing sign language proves challenging for individuals without specialized training in its subtleties.

Hand motion recognition through visual means has been a prominent technique in computer vision and machine learning. Given its inherent alignment with human interaction, researchers are trying to simplify and naturalize human-computer communication, eliminating the need for supplementary equipment. This effort emphasizes the fundamental objective of gesture recognition research—to create systems proficient in distinguishing diverse hand gestures for seamless communication. In the machine translation context, information retrieval and various other applications, such as tagging of Parts of Speech (POS), could be considered primary methods [6]. Furthermore, Deep Learning (DL) has a notable impact on improving the models to identify gestures. DL architectures establish connections beyond immediate neighbors within the data, thereby building learning patterns and formulating data-driven representations, all accomplished without human interaction [7].

Unlike the Western world, there has been little research on Bengali sign language. Studies on sign language words are rare; they generally focus on alphabetic and numeric symbols [8–11]. By placing something in its larger literary context, diverse works produced in languages other than English become apparent. We attempted to combine Bengali with contemporary technologies to mitigate the research gap. Our study represents a paradigm shift in research on sign language detection by offering novel identification methodologies. Bengali sign language detection is the exclusive focus of our work because past research has mainly focused on languages other than this one [12]. By creating a new dataset of frequently used Bengali words, we attempted to advance the Bengali sign language detection era. This study would advance the field of identifying sign language in Bengali, given that it is the seventh most spoken language in the world [13]. The results contribute to the current body of knowledge in sign language, which is also applicable in real-world settings, opening new avenues for further research and development.

This study would make a transformative contribution to the current research community working on Bangla sign language. First and foremost, it addresses a critical gap in the field by focusing on Bengali sign language, which has yet to receive much attention in research on sign language detection and recognition. This study would broaden the application of accessibility technology and develop a framework to create sign language recognition algorithms that would be useful to adapt to particular regional languages. Additionally, this research introduces identical computer vision and machine learning techniques optimized for sign language recognition, greatly enhancing the accuracy and efficacy of sign language interpretation. These developments could have a positive impact on a variety of applications, such as assistive technologies, instructional materials, and communication aids for those with hearing and speaking problems.

Since there has been extensive research on sign language detection and recognition in the Western world, our focus on Bengali sign language has been a timely shift. This study explores the distinctive nuances of Bengali sign language, as it has a unique vocabulary and grammar. We have developed an appropriate methodology to deal with its linguistic features. In essence, this study would establish the originality of Bengali sign language, advance our understanding of regional sign languages, and introduce novel approaches for their recognition. Key contributions of this study include:

- We are creating an exclusive dataset for this research endeavor, comprising nearly a thousand image data instances of commonly used BdSL words. Notably, the dataset focuses on ten frequently used words - Color, Friend, Myself, Promise, Request, Salam, Surprise, They, Think, and You.

- Developing a real-time word detection technique to identify the target words in practical contexts would lay the foundation for potential expansion into a comprehensive support system that meets the D&D community's particular communication needs.

The structure of this article is as follows: We will offer a concise overview of relevant literature on sign language in Part II and propose our research strategy in Part III. Part IV provides a brief explanation of the dataset. Part V offers the model description. We will then present the results of our study in Part VI and give a comparative analysis in Part VII. Finally, we will discuss the benefits, constraints, and future research scope in Part VIII and write a conclusion and references in Part IX and Part X, respectively.

2- Literature Review

Over the last few decades, many scholars have been involved in sign language detection research and developed various applications [14–16]. Language detection is crucial in bridging the communication divide between D&D individuals and hearing people. In the late 1990s [7], interest in sign language research grew, especially in the Western world. Since then, this discipline has rapidly expanded, giving rise to a growing corpus of research endeavors.

2-1- Research on Other Sign Languages

When considering other languages, one would notice that ample research has been conducted on American Sign Language (ASL), followed by British Sign Language (BSL) and Indian Sign Language (ISL).

Rahman et al. [7] have developed an enhanced CNN model to identify ASL. Their model improved the accuracy rate by almost 9%. They have worked on four available datasets and applied their proposed SLRNet-8 model to those datasets. After combining the alphabet and digit datasets, they achieved an overall accuracy of 99%. Sarawate et al. [17] proposed a model to identify real-time ASL sign language using neural networks. Physical components are used in this work to recognize the signs. A CyberGlove with 18 sensors and a Flock of Birds are used in this work to track different parameters of hands like bending, position, orientation, etc. Their system recognized isolated signs that achieved an overall accuracy of 92.5%.

Rajam et al. [18] studied Tamil Sign Language detection using image processing. They worked with the up and down positions of five fingers of a hand, converting them into 32 combinations of binary images. They created 320 images through a webcam. They used terminologies like palm image and feature point extraction, which achieved almost 96.87% accuracy. Stein et al. [19] described a data-driven method for translating sign language to speech for American and Irish Sign Languages. They have used terminologies like RWTH-Boston-104 and ATIS Corpus, in which the WER (word error rate) was 21.2% and 45.1%, respectively.

2-2- Research on BdSL (Bangladeshi Sign Language)

Compared to other sign languages, BdSL has been subject to limited research. Most work on BdSL has been on the alphabet and digits dataset. No worthy work is apparent on the signs representing words.

Hossen et al. [8] described Bangladeshi sign language detection using DCNN (Deep Convolutional Neural Network). They worked on 1147 images, categorized into 37 classes where each class defined a letter of the Bangla alphabet. They have fine-tuned the top layers of DCNN and achieved 84.68% accuracy on the validation dataset.

Himel et al. [9] proposed an ANN (Artificial Neural Network) and Computer-Vision-based Bangla sign language processing method. They collected pictures using Kinect (a combination of RGB camera, depth sensor, and multi-array microphone), and then respectively applied feature extraction, feature vector creation, and matrix creation, and finally fed it to the model for training. They have a success rate of almost 96%. Karmokar et al. [10] worked on a Bangladeshi Sign Language Recognizer by efficient NNE (Neural Network Ensemble). They worked on 235 images of 47 signs representing the Bangla alphabet and got an approximate accuracy of 93%.

Islam et al. [11] proposed a CNN (Convolutional Neural Network) model-based approach to identify signs of Bangla digits. They have created their dataset, which consists of 1075 images of all 10 digits, and used an 80:20 ratio for training and testing, respectively. They have achieved an overall success rate of 95%. Uddin et al. [20] represented an SVM (Support Vector Machine) based Bangla sign language detection system. They have worked on the Bangla alphabet. Their process started with converting the RGB (Red, Green, and Blue) images into HSV (for Hue, Saturation, and Value) color space. They used terminologies like Gabor filters (to acquire hand sign features) and Kernel PCA (to reduce the dimensionality). The SVM worked to classify the candidate features. Using the MATLAB tool on 2400 images, they have a success rate of 97.7%.

Podder et al. [12] used the deep machine learning models trained on two datasets (Alphabet and Numeric). They developed a real-time interpreter for Bangla Sign Language. The background, hand orientation, and skin tone were other focusing factors. The ResNet18 model outperforms all other models, according to the study. Muhammad et al. [21] proposed a Bangla sign language model algorithm for a 51-character system that combined alphabetic and numeric characters. Here, they utilized only 36 symbols to identify the joint-lettered letters. Bengali signs are typically used for vowels, consonants, and joint letters. Also, they frequently attempted to decipher secret letters from recognized ones.

This work stands out from prior research due to several distinctive features. Notably, there has been limited exploration for recognition of 'words' within the Bangladeshi sign language domain, as evident from the literature review. A scarcity of studies has focused on this specific facet of Bangladeshi sign language. It is evident in sections 2-1 and 2-2 that there has been research on sign languages, particularly ASL. It also indicates that there has been research on Korean, Tamil, Irish, British, and some Indian languages. We can observe that their strategy is based on deep convolutional neural networks in Bangla and other languages for sign language. In particular, the datasets of all earlier studies were based merely on the alphabet and digits. However, our data is centered on Bengali words. In addition, there is no such research that worked on real-time detection of Bangla sign language. Our system is reliable enough to detect and translate sign language in real time. Despite Bangla being the native language of over 300 million individuals, its sign language representation remains relatively under-explored. The differentiating factor between this study and its predecessors lies in the creation of a meticulously produced dataset that comes close to near perfection to enhance detection accuracy. The study showcases the application of ResNet50 and MobileNetV2 models to attain the highest accuracy level. Another unique part of this study is the application of multiple models—a departure from prior approaches. With the employment of over five models on the dataset, collectively producing consistently proficient and accurate outcomes, the reliability of the dataset is underscored. Furthermore, this research introduces a novel dimension by incorporating real-time detection, allowing the correct identification of BdSL words.

3- Proposed System

Figure 1 succinctly illustrates the concept underlying the proposed system. The process initiates with the collection of RGB-mode images. Upon ingestion into the system, the images undergo initial preprocessing followed by normalization. These processed images are systematically organized into datasets, forming the basis for subsequent analysis. The normalized image data is then partitioned and channelled into deep neural network (DNN) models for further processing. The convolutional blocks of the DNN models redeem the vital features of the image data and generate every node weight based on those features. The conclusive dense layer of the model comprises ten neural nodes and an activation function (SoftMax) to help determine the classification. In cases where satisfactory outcomes are not achieved, the system parameters are changed to different values to achieve optimal accuracy.
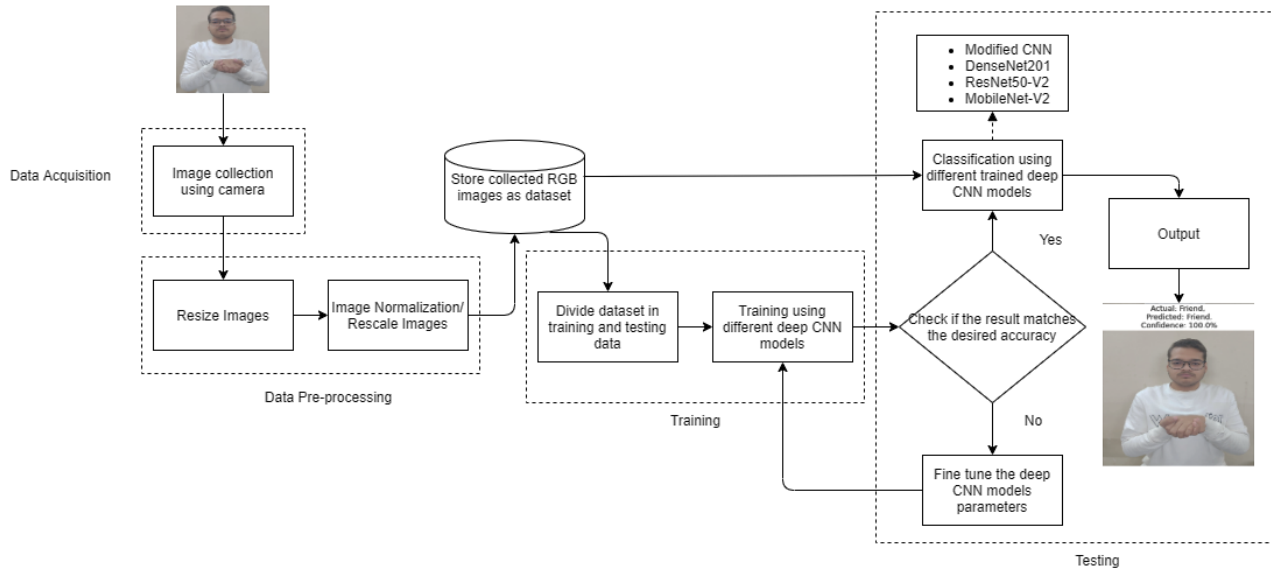
Figure 1. Proposed system

Following the acquisition of images from the dataset, the system encompasses different data preprocessing techniques, including rescaling, shearing, resizing, zooming, flipping, rotation, and more. Along with the customized CNN model, we have used three other DCNN (Deep Convolutional Neural Networks) models, which are ResNet50-v2, MobileNet-V2, and DenseNet201. The primary focus of this study is on feature extraction and subsequent classification from images, ultimately leading to outcome prediction through the implementation of the most fitting model.

4- Dataset Description

In the broader context of communication, signs serve as visual symbols to represent various linguistic concepts. Specifically, within the Bangladeshi Sign Language (BdSL), a distinctive form of communication utilizing gestures exists. These are a specific category of sign that holds particular significance. In alignment with the principles of BdSL, we have deliberately opted to focus on some of those signs. Within the paradigm of sign language, three fundamental categories emerged - words, numeric symbols spanning from 0 to 9, and alphabetic symbols (the latter being contingent upon the specific linguistic context). These delineated categories collectively encapsulate the breadth of expressive elements within sign language communication.

There is a noteworthy lack of easily accessible datasets that include BdSL terms because there haven't been any previous studies devoted to this field. Hence, we collected a dataset on the ten most used BdSL words. The "D&D People of Bangladesh" [22], which followed a thorough instruction manual, served as the basis for data accumulation. We created a dataset with ten selected words, each designated a distinct class. We gathered 100 photographs for each class, resulting in an overall dataset of 1000 images. This dataset was systematically partitioned into distinctive training and testing categories to facilitate a robust evaluation of our model's performance. For an in-depth understanding of the composition and characteristics of the dataset, refer to the detailed description presented in the subsequent table.

4-1- Data Overview

This table (Table 1) lists the quantity and snapshots of the test and training data and the number of photos for each of the ten classes in the dataset.

Table 1. Dataset overview according to each class

| Class names of the BdSL Words	Number of images in each class	Number of images for train	Number of images for test
Color (রঙ)	93	75	18
Friend (বন্ধু)	103	82	21
Myself (আমি)	105	84	21
Promise (কসম)	90	72	18
Request (অনুরোধ)	105	84	21
Salam (সালাম)	95	76	19
Surprise (অবাক)	100	80	20
They (তারা)	97	78	19
Think (চিন্তা করা)	106	85	21
You (তুমি)	98	78	20
Total	**992**	**794**	**198**

4-2- Data Sample

Figure 2 represents sample images for each class used in our work. Ten random images from the dataset are presented here for the classes color, friend, myself, promise, request, salam, surprise, they, think, and you, respectively.

Figure 2. Data Samples

4-3- Data Preprocessing

Data preprocessing constitutes a pivotal phase wherein a spectrum of morphological operations is applied to the data to mitigate noise and enhance quality [14]. In the context of our research, image acquisition was facilitated through webcams and cellular devices. Given the inherent diversity in image resolutions and sizes, a requisite step involved image adjustments to ensure compatibility with the models under consideration. Post-data collection, discrepancies such as inconsistencies, gaps, or unclassified elements may emerge, underscoring the need for data preprocessing. This facet of the research journey posed significant challenges and demanded substantial time investment. Notably, the diverse origins of the collected data necessitated the application of distinct data preparation methodologies. Despite these complexities, our team diligently endeavored to streamline the process, striving for efficiency. To achieve this, an array of data preprocessing techniques, including resizing, rescaling, shearing, zooming, rotation, and flipping, were employed, leveraging the suite of capabilities offered by the Keras framework.

The graphical illustration of Figure 3 shows how a scaled image previews. Because the photos in this dataset have different pixel sizes, we resized and converted them all to (224×224×3). We all entered them into an image data generator for a better outcome, and 20% zoomed them in.
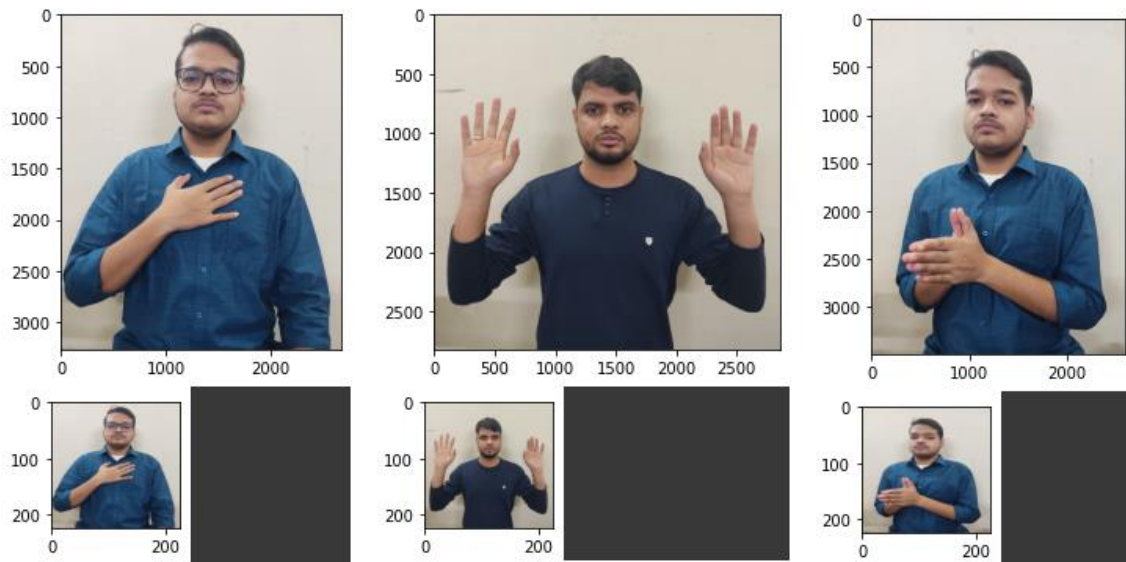
Figure 3. Resize of the images

As previously mentioned, the picture size we chose for the BdSL words dataset was (224*224*3), where 3 indicates that the image is RGB rather than greyscale. In addition, three subsequent 224*224 matrices will be produced if this image is converted to an array. Even though each matrix will have a significant value and be aggregated later in our algorithms to estimate the variance of each image, there is only one matrix when working with greyscale photos, as is well known.

Rescale: Rescale value was set to 1./255, which changed the value of pixels in the range of [0,255] to [0,1]. The input is normalized throughout this step (Figure 4 - Rescale).
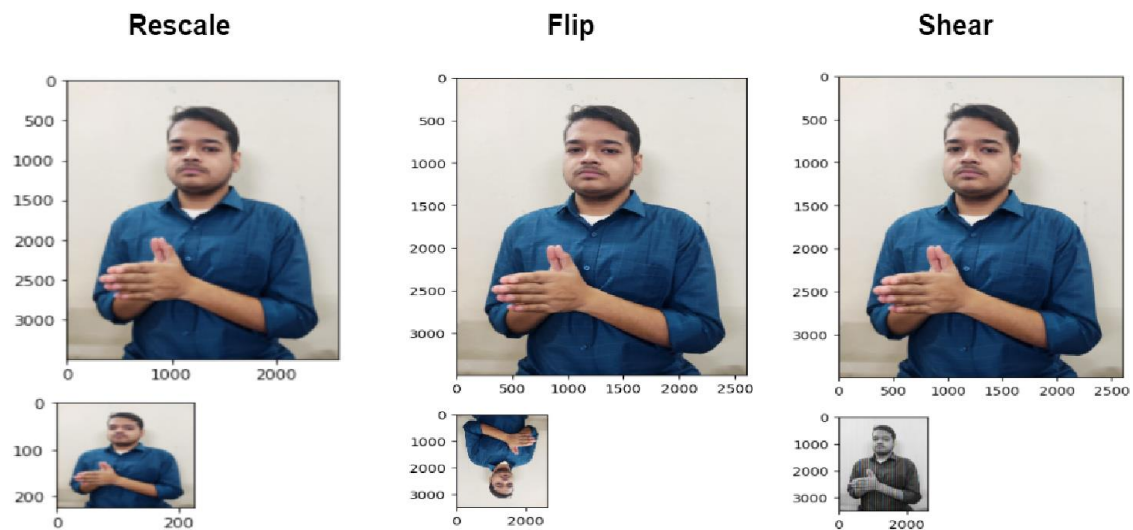
Figure 4. Image pre-processing methods (Rescale, Flip, Shear)

Flip: Two types of flip operations are applied. These are horizontal-flip and vertical-flip. These attributes randomly flip the bisection of the images horizontally and vertically, respectively. To create variation in the dataset, we also used parameters like "horizontal flip" and "vertical flip." Figure 4 - The flip portion shows the example of image flipping.

Shear: Shear angle refers to keeping one axis fixed while stretching the image at a fixed angle. This shift gives the images some "stretch" not visible during rotation (Figure 4- Shear). The shear range specifies the degree-level tilt angle.

These photos are then normalized after being resized to change the range of pixel intensity values, producing a mean and variance of 0 and 1, respectively. Image normalization, a standard method in image processing, adjusts the range of pixel intensity values. The term "normalization" relates to the process' usual action, which is to change the pixel values of the input image into a range that is more familiar or "normal" to the senses.

The following formula is used to normalize a digital image linearly:

$$output_{channel} = \frac{255*(input_{channel}-min)}{min-max} \tag{1}$$

As RGB images are used in this project, three channels were needed to normalize using the same technique.

5- Model Description

5-1- DenseNet201

DenseNet may be a better solution for picture classification in terms of lowering computing costs and obtaining strong gradient flows. DenseNet typically has four distinct variations. However, the highest accuracy of this work has been gained using DenseNet201, which is 93.4%. The DenseNet-201 is a portion of the CNN network with 201 layers; each receives additional input from all layers before it and then drives that input into all layers after it. Every layer in this system gathers information from the base layer. Because feature maps were collected from the previous layer, the network may narrow and have fewer channels. As a result, the number of additional channels has a growth rate of K for each layer. Figure 5 represents the structure of the DenseNet201 model.
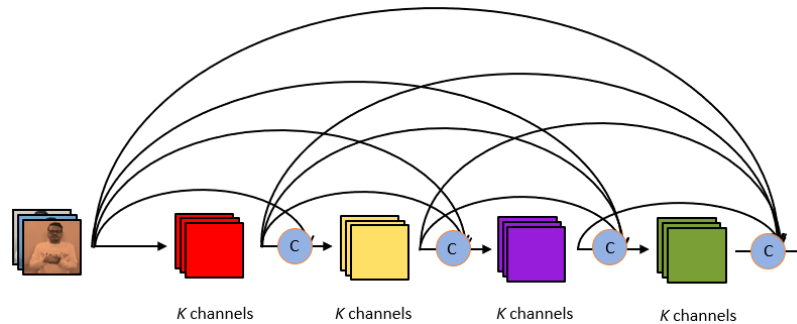

Figure 5. **Basic residual blocks of DenseNet-201**

5-2- ResNet50-V2

ResNet amplification is a Residual Network, a particular kind of neural network developed by Microsoft in 2015 that took first place in the ImageNet competition. A network using the ResNet network design is called ResNet50. ResNet is the driving force behind the revolutionary residual mapping approach, which solves the dirt issue. It has about 23 million trainable parameters and 50 layers. It supports RGB (Red, Green, and Blue) pictures. In contrast to other conventional DCNN designs, it uses extensive regular combining rather than ultimately linked layers [23]. Images are flawlessly detected and provide the highest accuracy of any architecture using this model. The network is initially dense by skipping just a few layers, enabling quicker learning. These layers are stretched, examining more of the feature in the remaining areas of the network as it reactivates. Figure 6 represents the structure of the ResNet50-V2 model.
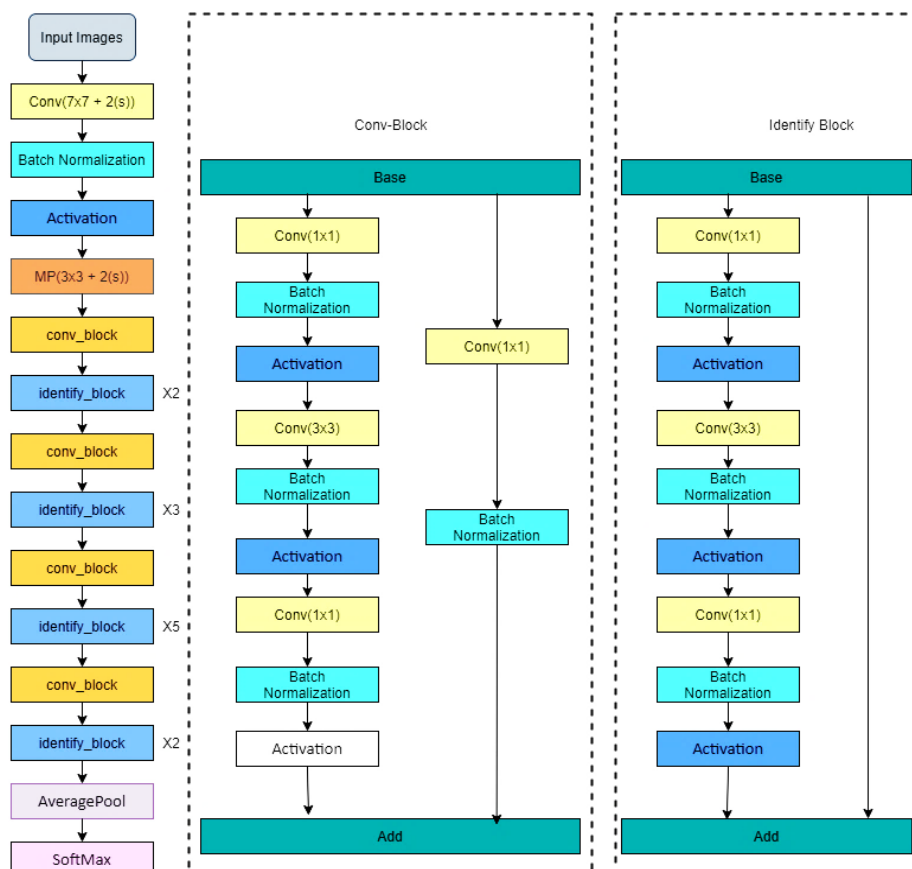

Figure 6. **ResNet50-V2 architecture**

5-3- Modified CNN

Objects in images can be located and categorized using convolutional neural networks. It is a feed-forward neural network. It also covers data processing utilizing a grid-like design for visual data presentation. This model significantly impacts the algorithm for deep learning. The three primary layers of the system are convolutional, pooling, and fully connected. It can examine data and predict events automatically. This computation offers solutions for problems with segmentation, classification, and image processing, among other things. Network weight initialization is essential because deep neural networks have unstable gradients, which could make learning more complex [24]. We know CNN can have several layers, combining a convolutional layer, a max-pooling layer, and a fully connected layer. In the applied model for our dataset, we have used three conv2D and max-pooling layers, followed by a fully connected layer. The fully connected layer consists of an activation function named "*SoftMax*" which transforms the output of a network into a probability distribution over classes of expected results.

A vector of n real numbers is normalized into a probability distribution with n probabilities proportional to the exponential of the input values by the *softmax* function. It takes as input a vector x of n real numbers. In other words, before applying *softmax*, some vector components may be negative or bigger than one and may not add up to 1. The components can be read as probabilities because they add up to 1 and are in the range (0,1) after *applying softmax*. Additionally, more input components will result in larger probabilities.

The following formula can define the SoftMax function:

$$\sigma(i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_i}} \tag{2}$$

Here, σ, x_i, e, e^{x_i}, $\sum_{j=1}^{n} e^{x_i}$ respectfully identify the SoftMax function, i[th] elements of the input vector "x," exponential function applied to each element of x and the sum of the exponentials of all the elements in the input vector "x." As the input vector contains the logits (raw scores generated by the output layer of the neural network), the exponential operation is applied to each logit to make them non-negative. Finally, the output of the SoftMax function represents the probability that the input image belongs to the ith class, given the raw score (logit) associated with that class. These probabilities are used for classification purposes, and the class with the highest chance is considered the predicted class for the input image.

Figure 7 represents the structure of the modified CNN model that we used for our work.
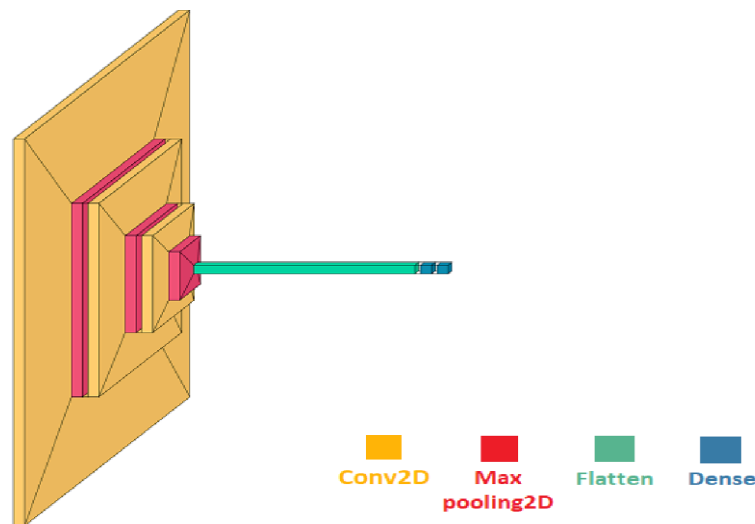
Figure 7. Modified CNN model architecture

5-4- MobileNet-V2

One of the most active and often used Convolutional Neural Network architectures in modern data science is MobileNetV2. The second MobileNet is an enhanced and advantageous version of the first. While the MobileNet-V1 algorithm either doubles or maintains the same number of channels, the MobileNet-V2 algorithm reduces the number of channels represented in Figure 8. The foundation of this model, which has 53 layers, consists of residual linkages between bottleneck levels and residual structure. This architecture provides real-time categorization capabilities and functions well on mobile devices. The first layer consists of a fully convolutional layer with 32 filters and 19 residual bottleneck layers subsequent. This setup allows a pre-trained network to classify over 1000 different item kinds. Typically, the image input size is 224*224 for MobileNet-V2 architecture.
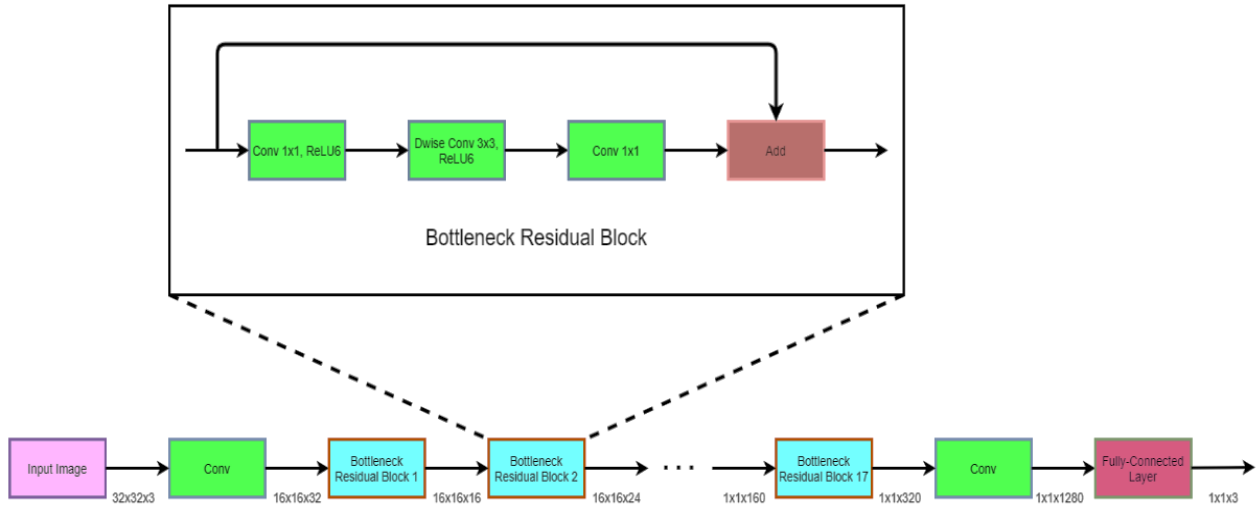
Figure 8. Residual block of MobileNet-V2

6- Result Analysis

6-1- Accuracy, Precision. Recall, F1 Score

This study examined the feasibility and precision of using Deep Convolutional Neural Networks (DCNNs) to recognize words in Bangladeshi Sign Language (BdSL). We aimed to close the communication gap for hearing-impaired people by applying advanced machine-learning techniques. Various deep-learning models were used, and the results were visually appealing and well-organized in recognizing BdSL words. As we examined the key results, we particularly focused on the accuracy of the models. All models' accuracy rates were very high. The accuracy of the four models, CNN, ResNet50v2, MobileNetV2, and Densenet201, ranged from 76% to 93%. Accuracy, recall, and F1 scores were also satisfactory and were frequently acknowledged. The TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative) were needed to analyze the overall accuracy and other results.

Accuracy: Accuracy usually refers to a ratio that displays the proportion of all the correctly predicted data. We can determine whether a model is appropriate based on its accuracy. If the FP (False Positive) and FN (False Negative) are the same, it helps to make high precision. The following formula can determine accuracy.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

Precision: It is a ratio that assesses positive observations. This metric suggests that it depends on how precisely predicted positive data is compared to all positive data. It displays a positive measuring ratio of genuine positive values, to put it simply. The equation is as follows:

$$Precision = \frac{TP}{TP+FP} \tag{4}$$

Recall (Sensitivity): In this context, the correctly predicted positive observations ratio refers to recall in situations where the response to the test question is "YES." The formula for calculating recall is as follows:

$$Recall = \frac{TP}{TP+FN} \tag{5}$$

F1 Score: The F1 score is a link between Precision and Recall, and in cases of unequal class distribution, the F1 Score is a more accurate statistic [15]. This term refers to the weighted average of recall and precision. Knowing what happened is beneficial when the cost of a false positive and a false negative is significantly different. As a result, both FP (False Positive) and FN (False Negative) values will be considered in this situation. The following equation gives the formula:

$$F1_{Score} = \frac{2*Precision*Recall}{Precision+Recall} \tag{6}$$

The following table displays each model's accuracy, precision, recall, and F1 Score (Table 2). The two models that give us the highest level of accuracy in our work are DenseNet201 and ResNet50-V2. The DenseNet201 model generated a performance table to show each class's accuracy, recall, precision, and F1 Score (Table 3)

Table 2. Statistical Analysis of Different Pre-Trained Keras Models

Model	Accuracy	Precision	Recall	F1 Score
CNN	0.76	0.77	0.76	0.76
ResNet50-V2	0.93	0.94	0.93	0.93
MobileNet-V2	0.87	0.90	0.86	0.87
DenseNet201	0.93	0.94	0.93	0.93

**Table 3. The performance for each class of Densent201 Model**

Classes	Precision	Recall	F1 Score
Color (রঙ)	0.95	1.00	0.97
Friend (বন্ধু)	0.95	1.00	0.98
Myself (আমি)	0.95	0.95	0.95
Promise (কসম)	0.90	1.00	0.95
Request (অনুরোধ)	0.79	0.90	0.84
Salam (সালাম)	1.00	0.95	0.97
Surprise (অবাক)	1.00	1.00	1.00
They (তারা)	0.89	0.84	0.86
Think (চিন্তা করা)	1.00	0.95	0.98
You (তুমি)	0.93	0.74	0.82

6-2- Accuracy and Loss Graph

The accuracy of each model and loss are presented in the preceding figures. Figure 9,11,13,15 represents the accuracy graph of all four models, and Figure 10,12,14,16 represents the loss graphs of all models. Here, the red line denotes accuracy/loss during training, and the blue line indicates accuracy/loss during validation. The number of epochs is presented along the X-axis, and the index of change in accuracy/loss is shown along the Y-axis.

6-2-1- Accuracy & Loss Graph of DenseNet201

The DenseNet-201 accuracy graph (Figure 9) shows that the starting point for training and validation accuracy is about the same. After completing a few epochs, it keeps improving, and five epochs of training accuracy remain in a constant range, which is very satisfactory. Due to data variance, validation accuracy varies slightly between different epochs, sometimes showing as high and other times a little bit less. We divide the data into two groups: 80%-20% (train-test), which the model chooses at random. Following some epoch data runs, the model was successfully trained and fit. Training accuracy reached over 100% after 25 epochs, and validation accuracy was 93%. The loss graph (Figure 10) for the DenseNet201 model initially has a very large scale; however, after fitting the data set, the loss for training and validation has practically reached zero. Because of this, the DenseNet201 model fits this dataset very well.
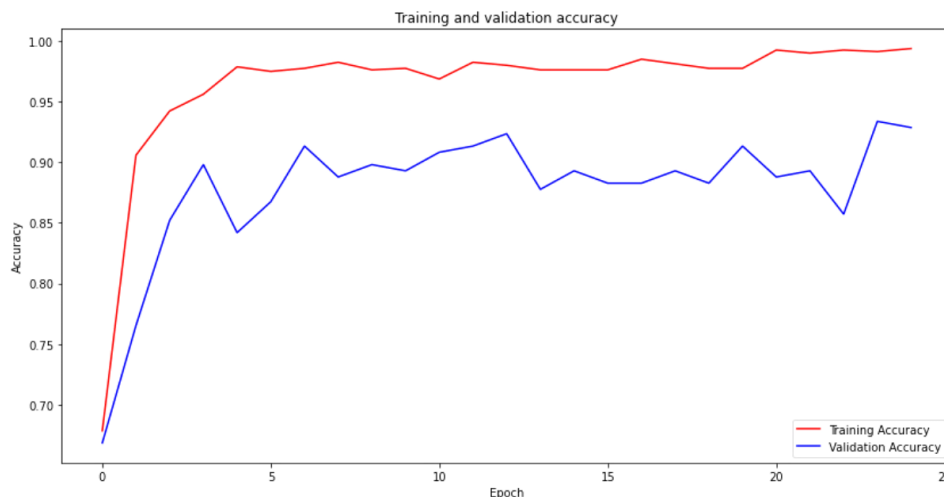

**Figure 9. Accuracy graph of DenseNet201**

Figure 10. Loss graph of DenseNet201

6-2-2- Accuracy & Loss Graph of ResNet50-V2

The accuracy graph for ResNet50-v2 (Figure 11) demonstrates that while there was an initial under-fit issue, this issue was fixed, and the initial accuracy increased after one epoch. After 7 epochs, the training accuracy has been consistent and practically linear; however, in epoch number 14, the validation accuracy is declining rather than the training accuracy, which is a concern due to the variation in the dataset. However, after 25 epochs, we obtained a satisfactory validation accuracy score of 93%. The training data loss is relatively little after one epoch in the loss graph of the ResNet50-V2 model, which is visible in Figure 12. The validation loss line varies significantly across different epoch numbers; however, sometimes, it is large across train data, and other times, it is close to each other.
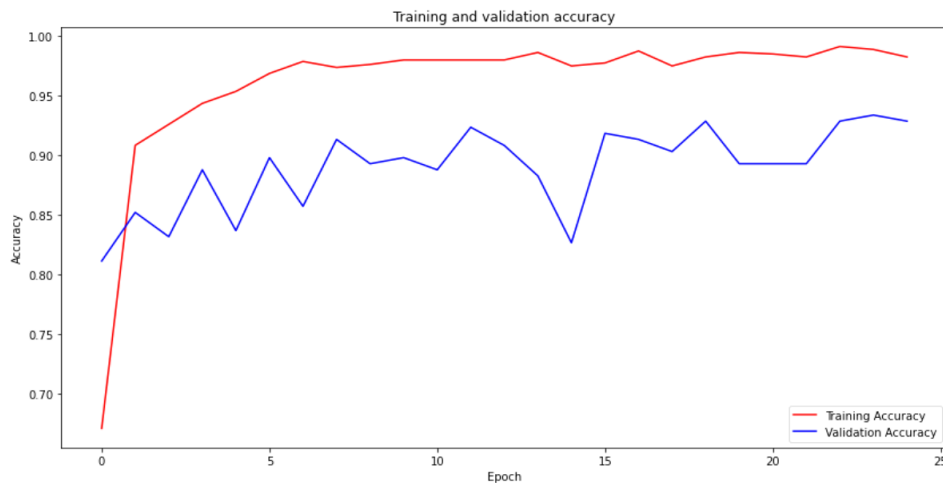
Figure 11. Accuracy graph of ResNet50-V2
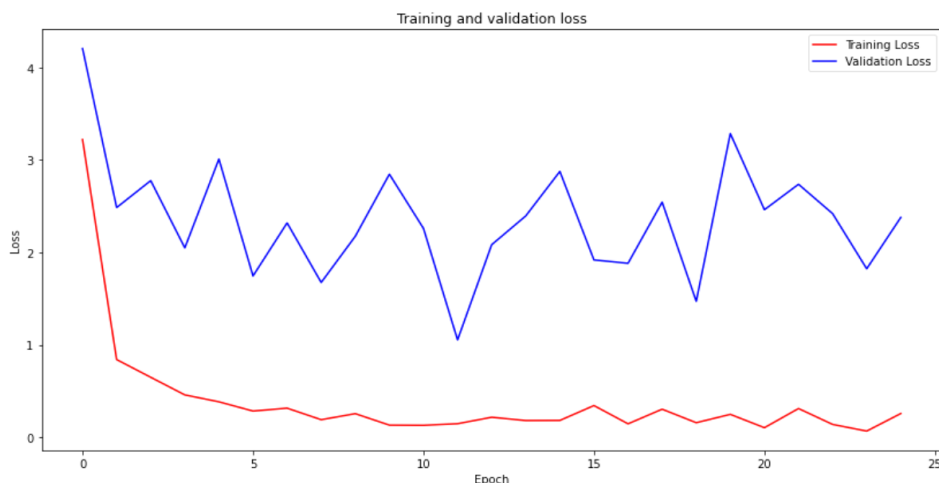
Figure 12. Loss graph of ResNet50-V2

6-2-3- Accuracy & Loss Graph of Modified CNN

For the CNN model, the accuracy graph (Figure 13) shows an under-fit problem before the third epoch. After that, training and validation accuracy increased significantly since the model fit well after a few epochs. And the over-fit problem is relatively low. There is a significant loss starting with training and validation data for Modified CNN (Figure 14). After the model has been fitted and some epochs have passed, the loss value decreases until the training loss is close to zero and the validation loss is between 1 and 1.5. In epochs nos. 6, 13, and 21, validation loss might occasionally be more significant than train loss.
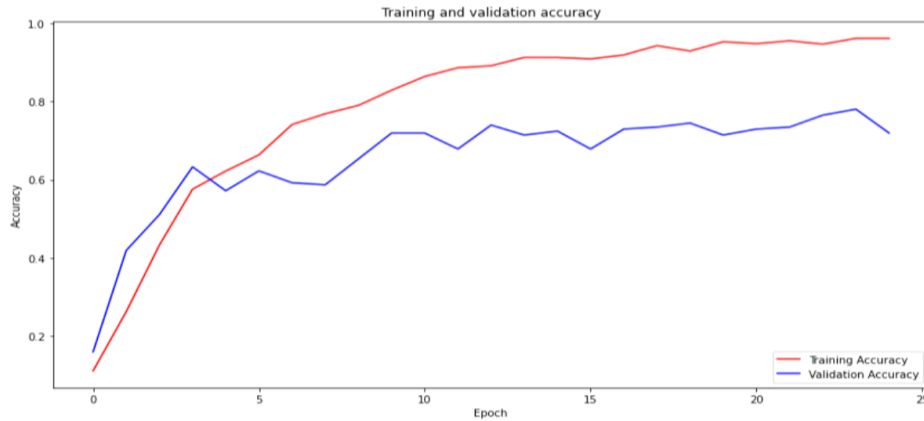

Figure 13. Accuracy graph of Modified CNN

Figure 14. Loss graph of Modified CNN

6-2-4- Accuracy & Loss Graph of MobileNet-V2

Figure 15 shows the training line in the MobileNet-V2 accuracy graph varies between 98% and 100%. And the validation graph line varies between 84% and 91%. In that situation, the overfitting problem at epochs 3 and 14 is obvious. Nonetheless, the model fit to the dataset usually provides higher validation accuracy than training accuracy. Finally, the accuracy of the model is 87%. There is almost no loss for training data in the loss graph of the MobileNet-V2 (Figure 16). However, the validation loss is significantly higher, which creates a difference between training and validation.
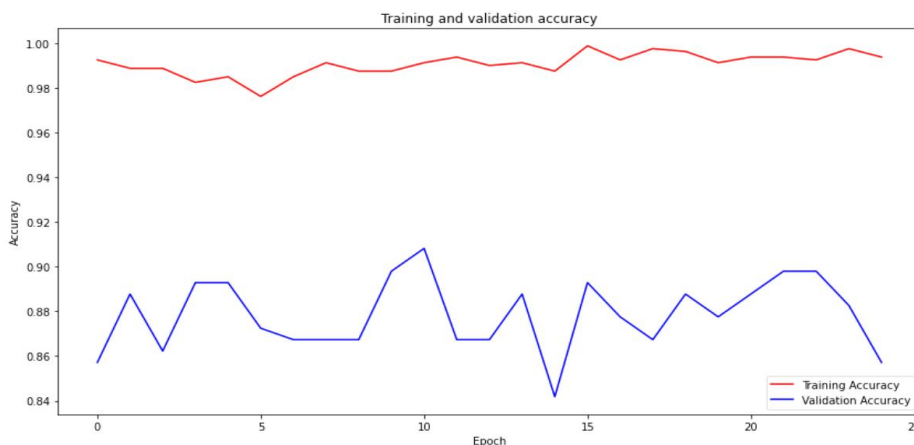

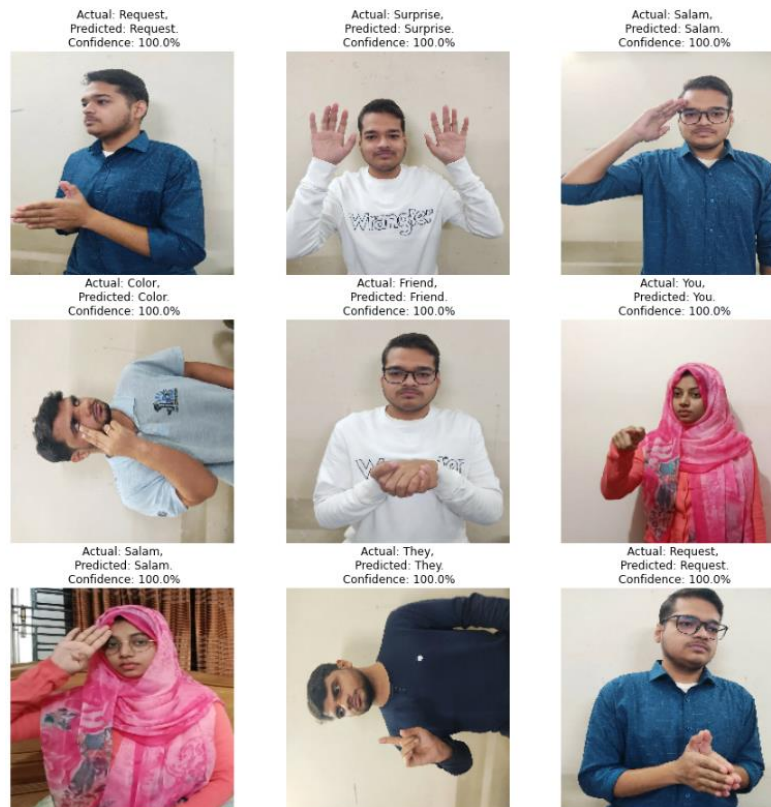**Figure 15. Accuracy graph of MobileNet-V2**

Figure 16. Loss graph of MobileNet-V2

6-3- Predicting Output

Here, we offer an example of forecasting output with Matplotlib using one of our most accurate models, the DenseNet 201 Model. Figure 17 displays the 100% precise outcome using nine different signs.

Figure 17. Predicting outputs using Matplotlib for DenseNet201 model

6-4- Confusion Matrix for the Models

The four key terms TP (True Positive), TN (True Negative), FP (False Positive), and FN (False Negative) stand for accuracy and error count, respectively, by looking at the confusion matrix for all the models. As a result, we are confident that our research is accurate. These four models most correctly predict ten different types of data. The X-axis of the Confusion Matrix displays the Predicted Class, while the Y-axis shows the True Class.

We employ 198 photos across 10 classes for the test, per the instructions in Table 1 of the dataset description. Color, surprise, and promise class forecasting accuracy is 100% in the confusion matrix of the DenseNet201 Model, while other classes, except the "You" class, have a 2-3 error maximum (Figure 18). For "You" Class TP- 14, FP-1, FN-5. In that case, we know this model may not be appropriate for this class (You). Maybe the visual data has complex or irregular structures or is likely to vary and be noisy. The Resnet50-V2 Model, one of the most accurate in our research, predicts the TP (True Positive) value for the classes color, Myself, Salam, and Surprise, with 100% accuracy. In contrast, the other classes have very few errors. For the You class, however, the TP (True Positive) value is also good, which is 15, the FP (False Positive) value is 1, and the FN (False Negative) value is 5 (Figure 19). The prior reason may have contributed to the error.
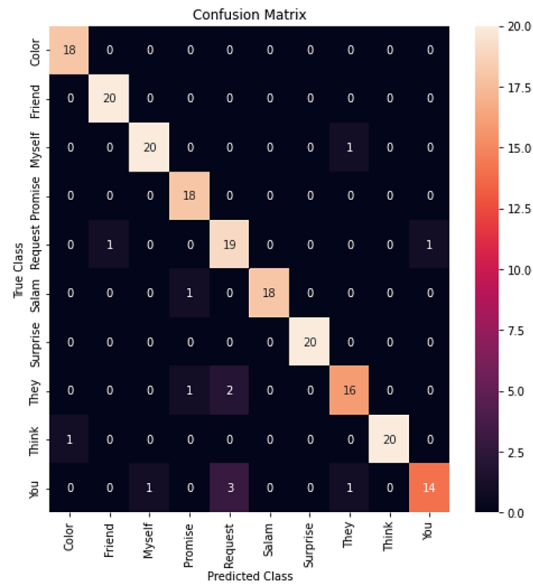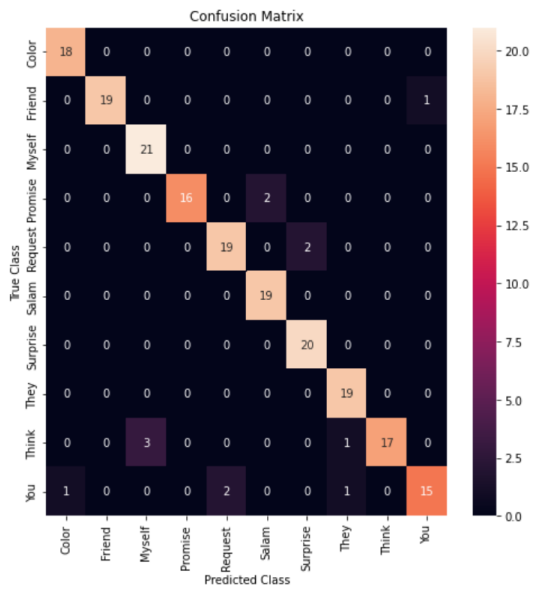
Figure 18. Confusion matrix of DenseNet201

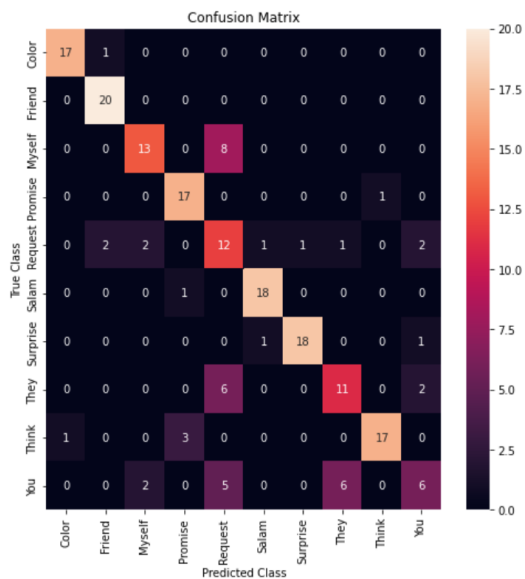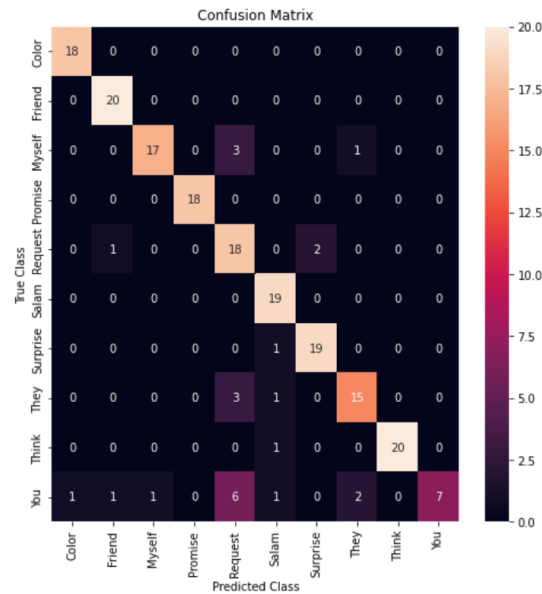**Figure 19.** Confusion matrix of ResNet50-V2

**Figure 20.** Confusion matrix of Modified CNN

Figure 21. Confusion matrix of MobileNet-V2

Although the TP (True Positive) values for our other two models, CNN and MobileNet-V2, are similarly highly excellent, the TP (True Positive) values for the You class are significantly lower than those for the other classes. According to the confusion matrix in class 'You,' the TP (True Positive) value of the CNN model is 6, the FP (False Positive) value is 5, and the FN (False Negative) value is 13 (Figure 20). However, mobileNet-V2 has 100% accuracy for the classes of Color, Promise, and Salam, and it also provides a wrong value for the Here You class, which is 6 for TP (True Positive) and 12 for FN (False Negative) however, 0 for (False Positive) FP- (Figure 21).

6-5- Real-time Detection

An illustration of a real-time test result that is nearly 100% accurate (Figure 12).
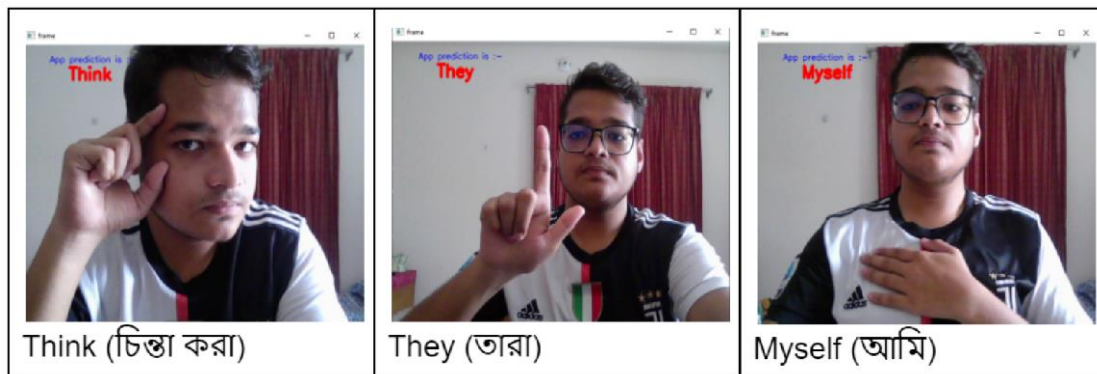
Figure 22. Real time detection sample for BdSL words

7- Comparative Analysis

As previously highlighted, more research on Bangladeshi sign language needs to be done. Few studies explored sign language words, with particular attention to alphabetic and numeric symbols. In the current study, we have attempted to incorporate modern technology with Bengali words and expressions. This study denotes a paradigm change in sign language detection research by providing new identification strategies. While earlier research focused on languages other than Bengali, this study has exclusively focused on the Bengali sign language detection process. We aimed to advance the Bengali sign language detecting period by compiling a unique dataset of frequently used Bengali words. Given that Bangla is a widely spoken language, our research would be impactful in this particular domain. Besides contributing to the sign language detection literature, this study would offer practical implications, paving the way for further research and advancement.

This section compares our contribution with the existing body of work. For a comprehensive understanding, we have conducted a systematic review of prior research, which has resulted in a comparative table (Table 4). This table is a crucial tool since it provides an analytical prism that can help us to identify specific qualities and differences of existing work.

Table 4. Comparative analysis with existing works

Reference	Dataset Types	Tools Technologies	Accuracy	Remarks
Rafi et al. [16]	Bangladeshi Sign Language alphabets	Pre-trained MobileNet-V2 model	94.74%	Worked On Bangladeshi Alphabets
Shin et al. [25]	Korean Sign Language words	Convolution and Transformer-based multi-branch network	89%	Worked on Korean words
Shamrat et al. [26]	Bangladeshi Sign Language numeric (0-9)	Convolution Neural Network (CNN)	99.8%	Worked on Bengali digits
Akash et al. [27]	Bangladeshi Sign Language (BdSL) alphabet, numeric, special characters	ImageNet, VGG19 Neural Network	99%	Applied YoloV4Tiny Model to recognize and LSTM to generate sentences
Lipi et al. [28]	Bangladeshi Sign Language (BdSL) words	Convolution Neural Network (CNN)	92.5%	Worked on Bengali words differently with lower accuracy
Hasan et al. [13]	Bangladeshi Sign Language (BdSL) digits and alphabets	Convolution Neural Network (CNN), KNN, SVM	95%	Worked on Bengali digits and alphabets
Sunanda et al. [29]	Bangladeshi Sign Language (BdSL) alphabet and digits	Deep Transfer Learning-based Convolutional-Neural Network with a Random Forest classifier	91.7%	Worked on Bengali alphabet and digits
Titarmare et al. [30]	Bangladeshi Sign Language (BdSL) words	Convolution Neural Network (CNN)	92.5%	Worked on Bengali words differently with lower accuracy
Our Research Work	**Bangladeshi Sign Language (BdSL) Words**	**Deep-Convolutional Neural and Transformer-based multi-branch network**	**93%**	**Worked on Bengali words with higher Accuracy and real-time detection**

Table 4 demonstrates that Rafi et al. [16] attained an accuracy of 94.74% using the mobileNetV2 model, although this only relates to the Bangla alphabet. Only a few studies have used Bangla digit and alphabet identification to attain enhanced accuracy. This table lists some research findings on the Bangla digits and alphabets.

The work on digit recognition by Shamrat et al. [26] needed more data. Hasan et al. [13] and Sunanda et al. [29] used the same type of Bangla digits and letters in the same circumstances. We noticed that their accuracy ranged from 91% to 99%. However, they only functioned with numbers and the alphabet, not with words. Another drawback of this work is the need for a real-time recognition mechanism. Akash et al. [27] used the Yolov4 Tiny Model to identify Bengali alphabet and numeric characters. While they attempted to detect each character accurately, their system focused on an IoT-based device and used many sensors, which is irrelevant to our work and research plan.

While examining the word recognition systems for other languages, such as Korean word recognition conducted by Shin et al. [25], we find that they only achieved 89% accuracy using convolutional Transfer-based multi-branch networks. Lipi et al. [28] studied ten distinct words frequently used in Bengali word recognition, but they produced little data and no real-time recognition. There are restrictions on Coollect's ability to collect a comprehensive dataset because it only permits classification in this scenario. Despite the lack of data, their model still achieved 92.5% accuracy.

Our research has produced a system for real-time recognition and has attained 93% accuracy - the highest accuracy in the most recent studies to recognize Bangla words. D&D persons can convey their words in real-time as text in both Bangla and English by using our system through a webcam.

8- Advantages, Limitations, and Future Scope

Bangla, one of the world's most widely spoken languages, has a user base exceeding 300 million who rely on it for direct communication. Most Bengali-speaking people live in South Asian countries like Bangladesh and India. Consequently, most of those who are D&D also live in these areas with significant difficulties when communicating. This research bears relevance not only to them but also to those who struggle with conventional communication. The study includes several frequently used Bangla words in everyday communication. The real-time accuracy attained in this study is impressive, and it offers significant trust in the accuracy of each word obtained by using dynamic deep CNN models.

Previous researchers have explored various sign languages, including Bangla; however, their focus has predominantly centered on datasets encompassing alphabetic and numeric symbols. However, this study brings a new epoch within this domain by introducing a different perspective. This approach has been remarkable because of its high accuracy and real-time detection capabilities. In particular, while distinguishing between individuals from the general population and D&D persons, this study excels in real-time detection with promising accuracy.

As this study has only focused on ten words in BdSL, one cannot define other words through this work. It is a limitation of this work, which in turn indicates the need for further research on other Bengali words. There is scope to include more commonly used terminologies from the actual world, making it more comprehensive. Besides, it can actively contribute to any gesture recognition work integrated with Human-Computer interaction, or HCI. The current research idea would assist in developing mobile or computer applications for communication, education, and accessibility. Through the use of these applications, people with hearing loss could be empowered and effectively communicate in a variety of settings. Another impactful sector for this work is facial expression recognition. In sign

language communication, facial expressions are essential for nuances in sign meaning. It would be possible to improve the depth and precision of sign language interpretation systems by incorporating facial expression analysis. There might be scope for future researchers to investigate a sign language learning platform using the concept of this research.

9- Conclusion

Irrespective of whether individuals are Deaf, Dumb, or both, they constitute an integral part of our society. By extending technological support to them, civilization could reach the pinnacle of its development. This study stands poised to bridge the gap between mainstream and differently-abled individuals. Bangla stands out as a crucial language among the many spoken languages, making the use of Bangla sign language in real-time essential. Notably, scant research has explored the Bangla sign language. The potential to accurately recognize signs in real-time through precisely provided predictors and dynamic deep learning models holds significant promise, which, in turn, could assist individuals facing communication challenges in surmounting societal barriers. The impact of our work is poised to be profound, fostering the establishment of a standardized community in the contemporary world through tangible implementation. The proposed methodology showcases enhanced accuracy in recognizing Bengali sign language, specifically tailored to an exclusive dataset crafted in alignment with the national curriculum.

Research on sign language is inadequate as it does not cover all relevant topics. We noticed in the brief literature review and result analysis that many languages require independent research for various sign language variables, such as digits, alphabet, special characters, words, and sentences. Most work is based solely on numerical data and a single alphabet or character.

So far, only limited research has focused on a few words. The low level of analysis, tiny dataset, and weak technique are their key drawbacks. In particular, researchers have yet to develop an organized system that could help recognize real-time Bangladeshi sign language as text while utilizing a webcam. Based on our dataset, which is a healthy and adequate amount of data, this study presents a dynamic system that offers real-time recognition and identification of the top ten common words.

Here, the complete working process, methodological concerns, and an evaluation of prior studies are all implemented constructively. The accuracy rises to 93% or higher across several models (ResNet50, denseNet201), a significant achievement thus far. The dataset is clean and reusable; further research within this area would save cost and time. The proposed methodology and the dataset could lead to further large-scale research to enrich the communication process using Bangladeshi Sign Language (BdSL).

10- Declarations

10-1- Author Contributions

Conceptualization, A.H.; methodology, A.H., R.A.P., and M.M.R.; software, R.A.P. and M.M.R.; validation, R.A.P., M.M.R., and S.A.; formal analysis, A.H.; investigation, R.A.P. and M.M.R; resources, R.A.P. and M.M.R.; data curation, R.A.P., M.M.R. and S.A.; writing—original draft preparation, R.A.P. and M.M.R.; writing—review and editing, A.H., S.H., and N.K.; visualization, R.A.P. and M.M.R; supervision, A.H.; project administration, R.A.P.; funding acquisition, A.H. and S.H. All authors have read and agreed to the published version of the manuscript.

10-2- Data Availability Statement

The data presented in this study are available on request from the corresponding author.

10-3- Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

10-4- Acknowledgements

We are grateful to the National Federation of the Deaf for releasing and enabling the people to utilize their documentary evidence on the conventional BdSL word signs. Besides, we appreciate the work of our earlier researchers, who sparked our interest in this topic. We respect the assistance of our coworkers in obtaining our data.

10-5- Institutional Review Board Statement

This study was approved by the Faculty of Science and Information Technology at Daffodil International University, Bangladesh.

10-6- Informed Consent Statement

An explanatory statement was given to the participants to read in which their rights and risks were outlined. After being satisfied with the statement, they signed the informed consent form.

10-7- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

11- References

[1] Wazalwar, S. S., & Shrawankar, U. (2017). Interpretation of sign language into English using NLP techniques. Journal of Information and Optimization Sciences, 38(6), 895–910. doi:10.1080/02522667.2017.1372136.

[2] BNFD (2023). The Deaf Annual Report. Bangladesh National Federation of the Deaf (BNFD), Dhaka, Bangladesh.

[3] Perlmutter, D. M. (1991). The Language of the Deaf. New York Review of Books, 28(3), 65-72.

[4] Perlmutter, D. M. (1993). Sonority and syllable structure in American Sign Language. Current Issues in ASL Phonology (Phonetics and Phonology), Academic Press, 227-261. doi:0.1016/B978-0-12-193270-1.50016-9.

[5] Chandra Karmokar, B., Md. Rokibul Alam, K., & Kibria Siddiquee, M. (2012). Bangladeshi Sign Language Recognition Employing Neural Network Ensemble. International Journal of Computer Applications, 58(16), 43–46. doi:10.5120/9370-3846.

[6] Blanco, E., & Moldovan, D. (2015). A Semantic Logic-Based Approach to Determine Textual Similarity. IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23(4), 683–693. doi:10.1109/taslp.2015.2403613.

[7] Rahman, Md. M., Islam, Md. S., Rahman, Md. H., Sassi, R., Rivolta, M. W., & Aktaruzzaman, M. (2019). A New Benchmark on American Sign Language Recognition using Convolutional Neural Network. 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh. doi:10.1109/sti47673.2019.9067974.

[8] Hossen, M. A., Govindaiah, A., Sultana, S., & Bhuiyan, A. (2018). Bengali Sign Language Recognition Using Deep Convolutional Neural Network. 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (IcIVPR), Kitakyushu, Japan. doi:10.1109/iciev.2018.8640962.

[9] Kayas, G., Hossain, S., & Hasan, M. (2016). Automatic Recognition of Bangla Sign Language Using Artificial Neural Networks (ANNS) For Deaf and Dumb to Bridge the Communication Gap. International Journal of Recent Advances in Multidisciplinary Research, 3(7), 1649-1654.

[10] Bloshchynskyi, I., Bahrii, H., Nanivska, L., Tsviak, L., Isaieva, I., Skyba, K., ... & Mishchynska, I. (2022). Gender Characteristics of Individual's Linguistic Behavior in the Context of Future Translators' Professional Training. Emerging Science Journal, 6, 199-208. doi:10.28991/ESJ-2022-SIED-014.

[11] Islam, M. S., Mousumi, S. S. S., Azad Rabby, A. K. M. S., Hossain, S. A., & Abujar, S. (2018). A potent model to recognize Bangla sign language digits using a convolutional neural network. Procedia Computer Science, 143, 611–618. doi:10.1016/j.procs.2018.10.438.

[12] Podder, K. K., Chowdhury, M. E. H., Tahir, A. M., Mahbub, Z. Bin, Khandakar, A., Hossain, M. S., & Kadir, M. A. (2022). Bangla Sign Language (BdSL) Alphabets and Numerals Classification Using a Deep Learning Model. Sensors, 22(2), 574. doi:10.3390/s22020574.

[13] Hasan, M. J., Nahid Hasan, S. K., & Alam, K. S. (2022). Deep Convolutional Neural Network-Based Bangla Sign Language Detection on a Novel Dataset. Machine Intelligence and Data Science Applications. Lecture Notes on Data Engineering and Communications Technologies, 132. Springer, Singapore. doi:10.1007/978-981-19-2347-0_13.

[14] Wadhawan, A., & Kumar, P. (2020). Deep learning-based sign language recognition system for static signs. Neural Computing and Applications, 32(12), 7957–7968. doi:10.1007/s00521-019-04691-y.

[15] Talukder, D., & Jahara, F. (2020). Real-Time Bangla Sign Language Detection with Sentence and Speech Generation. 2020 23rd International Conference on Computer and Information Technology (ICCIT), 19-21 December 2020, Dhaka, Bangladesh. doi:10.1109/iccit51783.2020.9392693.

[16] Al Rafi, A., Hassan, R., Rabiul Islam, Md., & Nahiduzzaman, Md. (2023). Real-Time Lightweight Bangla Sign Language Recognition Model Using Pre-trained MobileNetV2 and Conditional DCGAN. Studies in Autonomic, Data-Driven and Industrial Computing, 263–276. doi:10.1007/978-981-19-7528-8_21.

[17] Sarawate, N., Leu, M. C., & Öz, C. (2015). A real-time American Sign Language word recognition system based on neural networks and a probabilistic model. Turkish Journal of Electrical Engineering and Computer Sciences, 23, 2107–2123. doi:10.3906/elk-1303-167.

[18] Rajam, P. S., & Balakrishnan, G. (2012). Recognition of Tamil sign language alphabet using image processing to aid deaf-dumb people. Procedia Engineering, 30, 861–868. doi:10.1016/j.proeng.2012.01.938.

[19] Stein, D., Dreuw, P., Ney, H., Morrissey, S., & Way, A. (2007). Hand in hand: automatic sign language to English translation. TMI-07 - Proceedings of the 11th Conference on Theoretical and Methodological Issues in Machine Translation, 7-9 September, 2007, Skövde, Sweden.

[20] Uddin, A., & Chowdhury, S. A. (2017). Hand Sign Language Recognition for Bangla Alphabet using Support Vector Machine. 2016 International Conference on Innovations in Science, Engineering and Technology, ICISET 2016, Dhaka, Bangladesh. doi:10.1109/ICISET.2016.7856479.

[21] Rahaman, M. A., Jasim, M., Ali, M. H., & Hasanuzzaman, M. (2020). Bangla language modeling algorithm for automatic recognition of hand-sign-spelled Bangla sign language. Frontiers of Computer Science, 14(3), 143302. doi:10.1007/s11704-018-7253-3.

[22] National Centre for Special Education (1997). Bengali Sign Language Dictionary. Ministry of Social Welfare in cooperation with the Norwegian Association of The Deaf and Bangladesh National Federation of The Deaf, Dhaka, Bangladesh.

[23] Rafi, A. M., Nawal, N., Bayev, N. S. N., Nima, L., Shahnaz, C., & Fattah, S. A. (2019). Image-based Bengali Sign Language Alphabet Recognition for Deaf and Dumb Community. 2019 IEEE Global Humanitarian Technology Conference (GHTC), Seattle, United States. doi:10.1109/ghtc46095.2019.9033031.

[24] Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint, arXiv:1409.1556. doi:10.48550/arXiv.1409.1556.

[25] Shin, J., Musa Miah, A. S., Hasan, Md. A. M., Hirooka, K., Suzuki, K., Lee, H.-S., & Jang, S.-W. (2023). Korean Sign Language Recognition Using Transformer-Based Deep Neural Network. Applied Sciences, 13(5), 3029. doi:10.3390/app13053029.

[26] Mehedi Shamrat, F. M. J., Chakraborty, S., Billah, Md. M., Kabir, M., Shadin, N. S., & Sanjana, S. (2021). Bangla numerical sign language recognition using convolutional neural networks (CNNs). Indonesian Journal of Electrical Engineering and Computer Science, 23(1), 405. doi:10.11591/ijeecs.v23.i1.pp405-413.

[27] Akash, S. K., Chakraborty, D., Kaushik, M. M., Babu, B. S., & Zishan, Md. S. R. (2023). Action Recognition Based Real-time Bangla Sign Language Detection and Sentence Formation. 2023 3rd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), Dhaka, Bangladesh. doi:10.1109/icrest57604.2023.10070072.

[28] Lipi, K. A., Adrita, S. F. K., Tunny, Z. F., Munna, A. H., & Kabir, A. (2022). Static-gesture word recognition in Bangla sign language using convolutional neural network. Telkomnika (Telecommunication Computing Electronics and Control), 20(5), 1109–1116. doi:10.12928/TELKOMNIKA.v20i5.24096.

[29] Das, S., Imtiaz, M. S., Neom, N. H., Siddique, N., & Wang, H. (2023). A hybrid approach for Bangla sign language recognition using deep transfer learning model with random forest classifier. Expert Systems with Applications, 213, 118914. doi:10.1016/j.eswa.2022.118914.

[30] Titarmare, N., Vaidya, C., Meshram, R., Dongre, A., Jawale, P., Bambale, N., & Awachaat, O. (2023). Hand Sign Language Detection - Using Deep Neural Network. 2023 IEEE International Students' Conference on Electrical, Electronics and Computer Science (SCEECS). doi:10.1109/sceecs57921.2023.10063060.