A COMPARATIVE ANALYSIS BETWEEN SINGLE & DUAL-HANDED BANGLADESHI SIGN LANGUAGE DETECTION USING CNN BASED APPROACH

BY

Gourob Saha Surjo
ID: 191-15-2450
AND

Biplob Kumer Ghosh
ID: 191-15-2358

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Mohammad Jahangir Alam
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Md. Sabab Zulfiker
Senior Lecturer
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

FEBRUARY 2023

APPROVAL

This Project titled **"A Comparative Analysis Between Single & Dual-Handed Bangladeshi Sign Language Detection Using CNN Based Approach"**, submitted by Gourob Saha Surjo, ID No: 191-15-2450 and Biplob Kumer Ghosh, ID No: 191-15-2358 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on February 4, 2023.

BOARD OF EXAMINERS

Chairman

Dr. Touhid Bhuiyan
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dr. S. M. Aminul Haque
Associate Professor and Associate Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Dewan Mamun Raza
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner

4/2/2023

Dr. Shamim H Ripon
Professor
Department of Computer Science and Engineering
East West University

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Mohammad Jahangir Alam**, **Senior Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for the award of any degree or diploma.

Supervised by:

28.01.23

Mohammad Jahangir Alam
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:

29.01.23

Md. Sabab Zulfiker
Senior Lecturer
Department of CSE
Daffodil International University

Submitted by:

Gourob Saha Surjo
ID: 191-15-2450
Department of CSE
Daffodil International University

Biplob Kumer Ghosh
ID: 191-15-2358
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty God for His divine blessing making us possible to complete the final year project/internship successfully.

We are really grateful and wish our profound indebtedness to **Mohammad Jahangir Alam**, **Senior Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "*Machine Learning*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts, and correcting them at all stages have made it possible to complete this project.

We would like to express our heartiest gratitude to our co-supervisor **Md. Sabab Zulfiker, Senior Lecturer,** Department of CSE, for his kind help to finish our project and also to other faculty members and the staff of the CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discussion while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In order to help the vast majority of the population learn sign language, specialists today frequently apply machine learning techniques. This project seeks to create a model that can recognize the letters in Bangladeshi Sign Language (BdSL) using a deep learning approach. This study is a comparison between the recognition of single-handed and dual-handed Bangladeshi Sign Language. Dataset, KU-BdSL is selected to train the single-handed BdSL and we selected dataset - BDSL 49 for dual-handed BdSL. We suggested pre-trained models of CNN based approach for the purpose of detection and classification. 30 different alphabets from single-handed BdSL and 36 different alphabets from dual-handed BdSL could be classified and recognized by our CNN models. Three pre-trained CNN models were employed. VGG16 outperformed the others by a wide margin. It correctly identified single-handed gestures with 98% accuracy. Regarding the ability to recognize dual-handed Bangladeshi Sign Language, VGG16 once again outperformed competitors with an accuracy of 90%.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1
INTRODUCTION

1.1 Introduction

We are blessed with the ability to speak with and express ourselves to others thanks to our voice. We made various types of sounds and then formed words from those sounds. The ability to communicate with others is what keeps us going and growing in our lives. However, the deaf and mute cannot afford the luxury of communicating through voice. In the past, a person was referred to as a deaf-mute if they could neither talk nor utilize sign language but were both deaf. Both can perform multiple tasks because they are merely unable to talk or hear. The only thing that sets them apart is how they communicate with the general public. Hearing-impaired people can easily lead regular lives if there is a way for hearing people and deaf people to communicate. They can only converse with each other through sign language. Despite their hearing impairment, deaf people who understand sign language are fully able to hear and speak. To communicate with the general public and the deaf population, sign digits are also helpful for daily accounting.

A visual language, sign language employs hand movements, facial expressions, body language, and motions. Deaf persons typically express their feelings using a variety of hand gestures. The subject of identifying Sign Language by means of various approaches has been the subject of extensive investigation. Bangladeshi Sign Language (BdSL) is the Bengali sign language used in Bangladesh and Kolkata, India. Bangladesh's deaf and mute people use BdSL for daily and official communication. On February 1, 2009, Bangladeshi Sign Language (BdSL) was declared the official sign language of Bangladesh by Prime Minister Sheikh Hasina [1].

1.2 Motivation

On February 21, 1952, the whole world faced a rare incident in history, where the people of Bangladesh sacrificed their lives for our mother language. That’s why Bangla Language is our birthright and our pride. For people who have hearing and speaking disabilities, Bangladeshi Sign Language (BdSL) is their birthright. Among the Bangladeshi language-based minorities, the Bangla Sign Language user population is the largest.

More than 70 million deaf individuals live on the globe, and they communicate with others by using 300 distinct sign languages, according to the World Federation of Deaf [2]. Disabled Peoples' International Asia-Pacific (DPI/AP) reported on February 7, 2014, that Bangladesh is home to around 2.6 million deaf persons who speak Bangladeshi Sign Language (BdSL) as their native language [3]. Normal individuals find it challenging to communicate or comprehend sign language. Therefore, the ability to recognize Bangladeshi Sign Language might aid in bridging the communication gap between hearing persons and the deaf-mute community.

1.3 Objective

Sign Language is difficult to comprehend and learn. Years of practice are required to master sign language. On online platforms, there is little documentation or proper knowledge of Bangladeshi Sign Language (BdSL). We frequently find ourselves in an embarrassing situation among sign language users because we do not understand the meaning of the signs. The objective of this research is to use Artificial Intelligence to determine the meaning of hand signs. CNN will therefore be used for data classification in our strategy. Convolutional neural networks (CNN, or ConvNet) are a class of deep, feed-forward artificial neural networks that have been successfully used for the analysis of visual data. When compared to other image classification methods, CNNs use very minimal preprocessing. This suggests that the network learns the filters, which were manually created for conventional procedures.

In this research, we are using two different datasets, one for single-handed sign gestures and another for dual-handed sign gestures. The KU-BdSL: Khulna University Bengali Sign Language [4] dataset is a single-handed sign gestures image set. It contains 30 consonants. Images of 30 different hand signs have been used. The BDSL 49: A Comprehensive Dataset of Bengali Sign Language [5] dataset is a dual-handed sign gesture image set. It contains both vowels and consonants.

The classification of these images is done by the Convolution Neural Network (CNN). This approach is one of the finest deep learning approaches for image processing. CNN has been used successfully in the creation of robots with face recognition and self-driving cars with navigation of roads and traffic signs. We chose to use this technique in our research for these reasons.

1.4 Expected Outcome

We will be able to connect with sign language users thanks to this research. That is why we are using both single-handed and dual-handed hand gestures for our work. As previously mentioned, we used 30 consonants of the Bengali alphabet of BdSL from KU-BdSL for this study, and we will be able to detect those images with ease thanks to machine learning. Our study will be able to classify the images of 30 different single-handed images of the alphabets of Bangladeshi Sign Language (BdSL) [6].

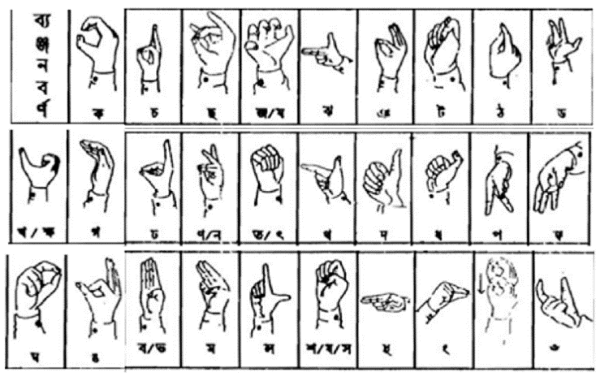

Fig 1.4.1: Single-Handed Sign Language (KU-BdSL).

For dual-handed hand signs, we used 6 vowels and 30 consonants of the Bengali alphabet of BdSL from BdSL 49. A total of 36 classes are used in this study to classify the images of 36 different dual-handed images of the alphabets [7] of Bangladeshi Sign Language (BdSL).

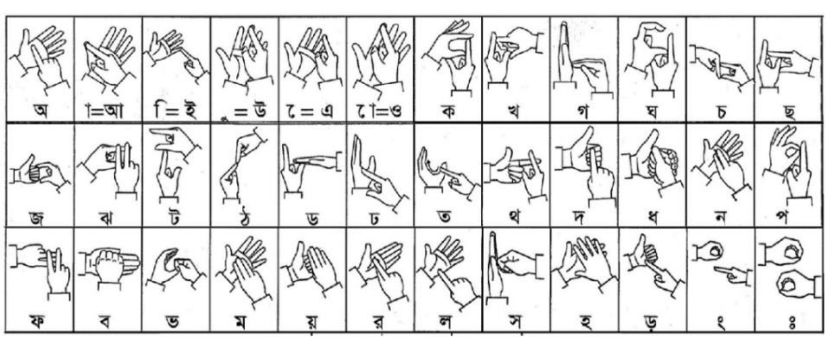

Fig 1.4.2: Dual-Handed Sign Language (BdSL 49).

1.5 Report Layout

- In the first chapter, we presented the essential concept of our research titled "A comparative analysis between single and dual-handed Bangladeshi Sign Language detection using ML-based approach" as well as our motivation, objective, and anticipated conclusion.
- In the second chapter, we focused on the works that are related to our study. We gave a summary of this study and also discussed the problems and challenges during the study.
- Methodology of the research is discussed in the third chapter. Such as data collecting, data processing, model training, model architecture, etc.
- The experimental outcomes and analysis of these outcomes were presented and discussed in Chapter 4.
- The final chapter is about the evaluation and conclusion of our work. We also discussed some extra features that will help us in our further study.
- At last, we included References, a Plagiarism report, and a list of figures.

CHAPTER 2

Background Study

2.1 Introduction

Gesture and sign language recognition is a cutting-edge academic topic. It may utilize several techniques, including sensory devices, to determine what the gesture is. However, in practice, using hardware is more expensive and inconvenient. Therefore, researchers are working to achieve the highest recognition accuracy using computer vision techniques. We explored and further discovered some research works where depth photos were utilized along with various methodologies for various types of research, taking the significant training loss into mind.

2.2 Literature Review

The goal of this research is to develop a model capable of detecting BdSL alphabets. Several scholars, each working in a different field, have explored a variety of ways of identifying different indicators.

Deep Convolutional Neural Networks are used in a novel technique for Bengali Sign Language Recognition (DCNN) [7][8]. This method is intended to recognize 37 static hand movements corresponding to the Bengali alphabet. They utilized 31 photo sets to create 37 unique signs to test with, and for each test they varied the accuracy of the features used to identify each sign. Using deep convolutional neural networks, learning features from a pre-trained network, and fine-tuning this network's top layers, they achieved a high overall recognition rate of 96.33% on the training dataset and 84.68% on the validation dataset.

Different types of Indian Sign Languages were proposed to be categorized using Eigen Value Weighted Euclidean Distance [9]. The four parts of the system are skin filtering, hand cropping, feature extraction, and classification. It was shown that 97% of the 240 images studied could correctly identify one of twenty-four different indicators.

CNN-based object detection method to find signs in an image region and identify their type [7][10]. The study used a Faster Region-based Convolutional Network technique for this aim and created the BdSLImset dataset to train the system. The experimental results demonstrate that the proposed technique, although lacking these constraints, is able to identify and detect Bangladeshi signals in real-time.

A research claims that smaller networks, such as LeNet-5 and AlexNet [11], and deeper networks, such as Vgg16 and MobileNet v2, are used to train the database from scratch. The report discusses a comparison of different recognition accuracies. The chosen design only includes 10 layers, one of which is a dropout layer that increased testing accuracy to 87.5% and training accuracy to 91.37%.

In a study, a feed-forward ANN with a backpropagation learning technique was applied [12]. Feature extraction began after the RGB images were converted to binary and the binary image was cropped to the hand region. The alphabet and numerals are among the 46 Bangali signs the system was designed to recognize. An Intel Core i3 computer running Windows 7 64-bit was used to run MATLAB 2013a. The proposed fingertip finder algorithm successfully located the sign's fingertip without the aid of a glove, sensor, or marking device. In comparison to other techniques, the accuracy of this system in recognizing BdSL is quite promising.

The fingertip is located using a cusp detection analysis and a boundary-trace-based finger detection technique [13]. This algorithm allows us to recognize a group of hand movements from American Sign Language that include open fingers by using a straightforward, effective, and reliable way to find fingertips. The Canny algorithm employs an ideal edge detector that takes into account a number of variables, including discovering the most edges while reducing error rates [13][17].

A research claims that smaller networks, such as LeNet-5 and AlexNet [11], and deeper networks, such as Vgg16 and MobileNet v2, are used to train the database from scratch. The report discusses a comparison of different recognition accuracies. Two extra layers are added to eliminate some parameters while still maintaining the fundamental characteristics of images to order to solve the overfitting problem. Additionally, the dropout layer, batch normalization, and data augmentation are implemented to get around the issue and improve our network's performance.

In this research, we employ a convolutional neural network and a learning-based approach to detect Bangla Sign Language (BdSL) [7][14][15]. Our proposed method [15] utilizes the Leap Activity Device (LMC), a VR-based hand-tracking controller, to monitor the hands' incessant motion. LMC provides a skeleton hand model complete with data on hand position, orientation, rotation, fingers, grasping, and other non-linear features.

This research suggests a BdSL system that trains ANN using the fingertip position [12][16][17]. It has been tested on large-picturere dataset and compared industry-standard approaches. It is shown that the proposed method outperforms the state-of-the-art techniques based on test set accuracy.

2.3 Comparative Analysis

Bengali Sign Language Recognition is one of the rarest of sign language recognition. We have discovered a small number of studies that discuss BdSL recognition. Numerous sign languages exist throughout the world. The sign language with the most research is American Sign Language (ASL), which is also the most well-known. Regarding sign language, we have been taught in several ways. Some of the approaches are very intricate for a basic and simplistic environment. For proper execution, they require high-quality hardware. However, there are studies that have presented us with simpler machine learning and deep learning techniques for conducting this type of study.

2.3.1 Alphabet recognition using Convolutional Neural Networks with multi-view augmentation and inference fusion in American Sign Language

Deep learning requires a lot of data, which takes time for CNN. To save time, data augmentation synthesizes more original data. Rotation, scaling, and keeping recognized elements intact provide more data variations [18]. This method captures a point's 3D cloud using a camera and virtual cameras. Each axis rotates yaw-pitch-roll around the volume center. CNN's model is regularized. Some model gestures are comparable to inter-class gestures. Multi-view inference fusion was proposed to enhance each view's hypothesis. Preprocessing entailed locating the hand section of the image, which was small.

The palm point is a circular region determined using the depth image's mass center. Overall process accuracy was 88%. Using 32831 testing datasets, half validation method accuracy was 99.9%. This strategy is useful for big data.

2.3.2 Microsoft Kinect Alphabet Recognition for American Sign Language

ASL alphabet identification using marker-less vision sensors is difficult due to the complexity of the signs, hand self-occlusion, and sensor resolution. This research describes ASL alphabet recognition using Microsoft's Kinect [19]. Using a depth contrast-based per-pixel classification technique, a segmented hand configuration is generated. A hierarchical mode-seeking approach is proposed to localize hand joint positions under kinematic restrictions. Using joint angles, a Random Forest (RF) classifier recognizes ASL signals. For validation, we used a Surrey University dataset. Our technique can recognize 24 static ASL alphabet signs with above 90% accuracy, substantially higher than prior standards.

2.3.3 Symbolic Gesture Recognition Using Depth Information

This study describes a symbolic hand gesture recognition system and uses depth map data effectively. This research uses depth value to determine grey-scale levels for contrast-varying depth images [20]. Hand-finger context information of the gesturing hand represented by local invariant feature descriptors has contributed to 96.84% identification accuracy, which is superior to binary images, images with edge information, and time-series curves. Preparing hand depth silhouettes affects gesture recognition accuracy. With depth map data, we produced rapid and effective gesturing visuals. Also essential is cluster size. Our empirical data show that 1600 clusters are optimal for accuracy. Pictures having solely edge information or binary images have fewer key points than depth images, hence they have more clusters. We took enough training samples to create the cluster and SVM classification models.

2.3.4 A Powerful Convolutional Neural Network Model for Recognizing Bangla Sign Language

Sign Language is difficult to comprehend for most people. Daily, deaf individuals use sign language to communicate. Sign language includes sign digits. So, they need a machine translator to communicate [7]. Computer vision-based technologies are popular for making technical language intelligible.

This project tries to build a deep-learning model to recognize BdSL digits. In this method, 12 signs were trained with a training dataset (Ishara-Lipi) using Convolutional Neural Network (CNN). The model was trained with 860 and evaluated with 215 test pictures. The training model recognized Bangla sign language digits with 95% accuracy. This model will advance BdSL machine translation.

2.4 Scope of the Problem

2.6 million Bangladeshis are deaf or mute. They communicate entirely using sign language. Most people do not use sign language. Having an interpreter is costly and inconvenient. We built a method to translate sign language to Bengali, taking into consideration the complexity and limitations. So that folks with and without disabilities can converse. Bengali has not been the subject of the extensive study that has gone into the detection of sign languages like American Sign Language (ASL), Canadian Sign Language (CSL), etc. There is no structured Bengali sign language dataset available. The dataset contains training errors and limitations. Therefore, this paper has the potential to be an excellent study. Following an extensive study, we have chosen two datasets. The KU-BdSL is chosen for one-handed BdSL, whereas the BDSL-49 is selected for two-handed BdSL.

2.5 Challenges

The first obstacle is locating suitable datasets. Finally, we located the datasets required for this study. After then, the most difficult aspect was preparing the environment for the study. We must utilize many machine learning library types, including TensorFlow and Keras. It was difficult to match Python's version with the versions of those libraries. However, machine learning and deep learning demanded robust hardware. To train our model, we used an AMD RX570 GPU. However, this resulted in a significant setback. The majority of Python libraries do not support AMD graphics cards. CUDA is the name of their parallel computing platform. It only supports graphics cards from NVIDIA. Due to the GPU crisis, we're unable to manage one. We attempted to install our AMD graphics card manually. In most cases, it crashed during model training. Therefore, we had no choice except to adopt Google CoLab. It gave us the boost we needed to effectively train our models. But because of the electrical crisis and internet issues, training the models was extremely difficult.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

The dataset is the most crucial part of training a machine learning model. We built two separate data sets by mining information from a wide variety of sources. To use these datasets in our research, we will need to clean and transform them into a format that can be utilized for training in a variety of models. After formatting or preprocessing is complete, we send the information for training.

3.2 System Design

This diagram illustrates our implementation system's design. This will provide an overview of the system we followed as shown in Fig 3.2.1.

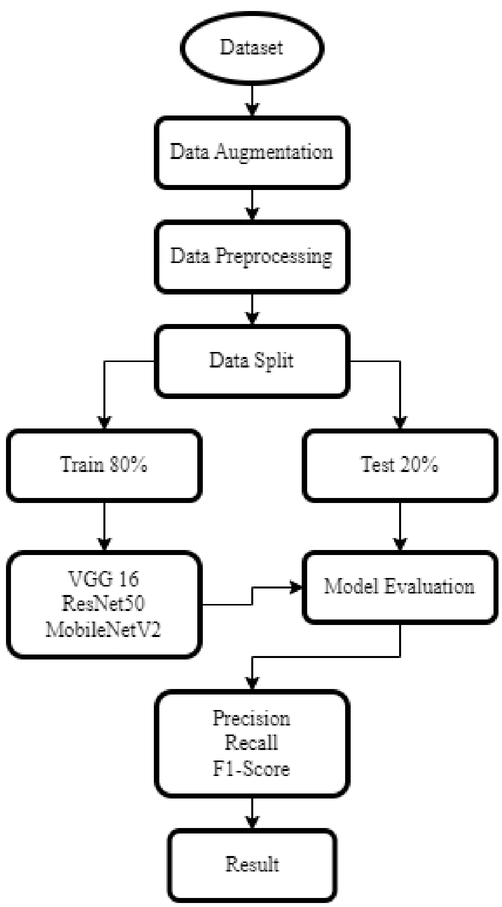

Fig 3.2.1: System Design for Implementation

3.3 Data Collection Procedure

We have collected two distinct datasets, one for single-handed sign gestures and the other for dual-handed sign gestures, which are used in this study. An image set of single-handed sign gestures can be found in the KU-BdSL: Khulna University Bengali Sign Language dataset [17]. There are 30 consonants in it. 30 different hand signals have been represented by images in KU-BdSl. A collection of dual-handed sign gesture images can be found in the BDSL 49: A Comprehensive Dataset of Bengali Sign Language dataset [18]. Both vowels and consonants are present at BDSL 49.

There are a total of 1,500 pictures in the KU-BdSL collection. Based on the consonants used in Bengali Sign Language, the photographs were sorted into 30 folders, with 50 images in each. All images were 512 pixels on a side and saved as ('jpg').

From BDSL 49, we extracted 36 folders containing Bengali Sign Language consonants and vowels. The number of images was 10814 in total. In the training set, each folder contains approximately 240 images, while in the evaluation set each folder contains 60. The images were all in ('jpg') format and varied in size. Eventually, we adjusted the size through pre-processing.

3.4 Manual Processing

The KU-BdSL dataset contains all consonants of BdSL within 30 categories. This set lacks a separate train or evaluation set comparable to BDSL 49. We manually enhanced some of the images using various image editing programs. Then, we developed two distinct sets, one for training and the other for evaluation. Each folder in the training set contains 100 images, for a total of 3000 images. The evaluation set contains 1,500 images, with 50 images per folder.

3.5 Augmentation

KU-BdSL consists of single-handed Bengali Sign Language. But This dataset doesn’t have enough data as BDSL 49. We manually created some data but had a great chance of overfitting during the model training. To remedy this, we employed the data augmentation method to expand the size of KU- BdSL's training set. Changes made to the pictures' spatial features yield new data. After the data augmentation, we were able to create 12000 images in the training set of KU-BdSL. 30 folders were able to have 400 images each.

Table 3.4.1: Applied Augmentation in KU-BdSL

No	**Augmentation**	**Tuning**
1	Rotation_Range	0.8
2	Width_Shift_Range	0.2
3	Height_Shift_Range	0.2
4	Shear_Range	0.02
5	Horizontal_Flip	Ture
6	Fill_Mode	reflect

3.6 Split the Data

Before training the models, we divided the training set from KU-BdSL and BDSL 49 into training and testing sets of 80% and 20%, respectively. Additionally, we created an evaluation set beforehand. Fig 3.5.1 and Fig 3.6.1 shows the number of images after the split.

```
Total number of symbols:  30
Number of training images:  9600
Number of testing images:  2400
Number of evaluation images:  1500
```

Fig 3.6.1: Data Split for KU-BdSl

```
Total number of symbols:  36
Number of training images:  6923
Number of testing images:  1731
Number of evaluation images:  2160
```

Fig 3.6.2: Data Split for BDSL 49

3.7 Data Pre-processing

Data preprocessing is one of the procedures required for training the data. Because not every image will have the same dimensions. Regarding model training, we must extract from each folder the labels that indicate the various Bengali Sign Language alphabets. From KU-BdSL, we obtained images of size 512*512, while BDSL 49 contains images of various sizes. Therefore, the photos are resized to 224*224. Consequently, all the photos can be adequately trained.

3.8 Model Configuration

As one of the most well-known methods to improve the precision of image categorization, Convolutional Neural Networks have become increasingly important in recent years (CNNs for short). We know that CNN (Convolutional Neural Network) is the largest deep artificial intelligence utilized for image processing. Named as such because it employs a convolutional mathematical process, convolutional neural networks are becoming increasingly important in the field of AI technology. A convolutional neural network is an artificial intelligence program that accepts images as input, assigns learnable weights to various visual objects, and is capable of classifying the images according to their respective categories. CNN resembles neural networks. Neurons have learnable weights and biases. Neuronal connection resembles brain architecture. CNN's goal is to compress images so they're easier to process without losing crucial features. CNN chose this strategy because pre-processing is minimal.

3.8.1 Convolutional Layer

It constitutes the fundamental building pieces of a convolutional neural network. These layer parameters construct a set of kernels or learnable filters, a limited receptive field that yet extends to the input's full depth. Not only is it utilized for input data and pixel values, but it can also be used for the output of other layers.

3.8.2 Max Pooling

Neuron sizes are decreased with the help of the Max Pooling layer. After each Convolutional Layer, this is the next step that is executed. The main rationale for using this layer is because it reduces the amount of processing power needed to process the data through dimensionality reduction. Maximum pooling provides the highest possible value for the filtered region of a picture.

3.8.3 Fully Connected Layer

Convolutional neural networks conclude with a fully connected layer. This layer is connected to all neurons that came before it. This layer's primary aim is to receive the results of the preceding convolution and pooling layers and use them to complete the classification process.

Incredibly complex as a tool, TensorFlow simplifies the creation, training, and use of Machine Learning models. Simulations of learning On our dataset, we use a CNN Model to analyze images. A monetary amount is provided below. Convolutional neural networks (CNNs) are a subset of deep neural networks used for display purposes. Typically, neural networks are employed to assess visual images [22]. All of the neurons in one layer are connected to all of the neurons in the next layer in a layered perceptron.

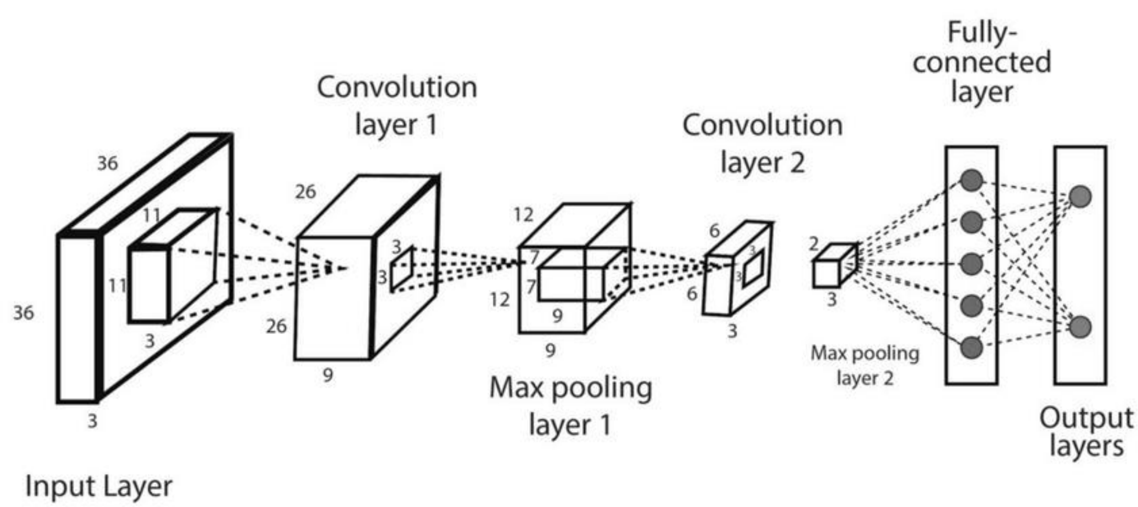

Fig 3.8.1: Basic Structure of CNN

3.9 Model Description

Machine learning techniques can be transferred from one dataset to another using transfer learning. This method is typically employed when there is insufficient data for thorough model training. That's why it makes sense to take advantage of a model that's already been trained along these lines. This work made use of models like Vgg16, ResNet50, and MobileNetV2 that had already been trained on the massive Image Net dataset, which includes millions of photos from a thousand different categories. In this study, we utilized three models (VGG16, ResNet50, and MobileNetV2) to determine which model delivers the highest degree of accuracy. In both of our datasets, VGG16 provides the highest accuracy, thus we choose to keep it as our final model to expedite the detection process, despite the fact that there is no better choice for achieving the highest accuracy for our proposed model. The VGG models are gaining support from an ever-expanding audience.

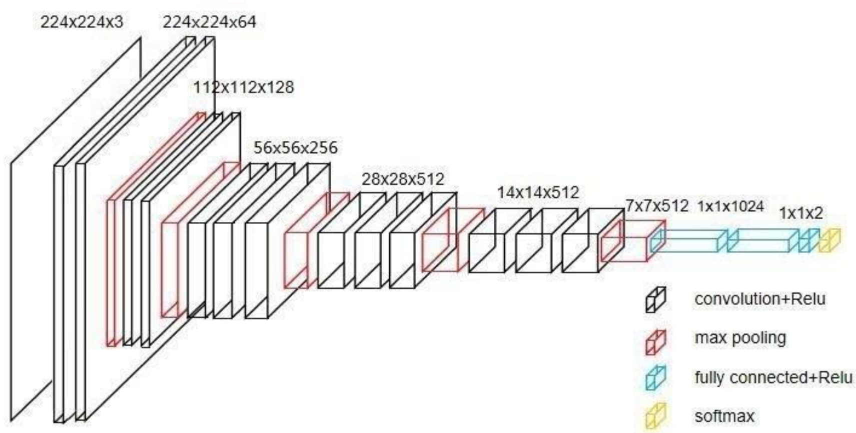

Fig 3.9.1: Architecture of VGG16

The VGG16 CNN architecture [23] triumphed in the 2014 ILSVR(ImageNet) competition. It's one of the most impressive vision model architectures ever created, and it's also the most precise model we've suggested. VGG16 has 16 weighted layers, as indicated by the number 16. There are around 138 million parameters in this large network. VGG16 avoids using a large number of hyper-parameters by instead concentrating on 3x3 filter convolution layers with stride 1, while consistently employing the same 2x2 filter stride 2 paddings and max pool layer. The structure of VGG16 is seen in Fig 3.9.1.

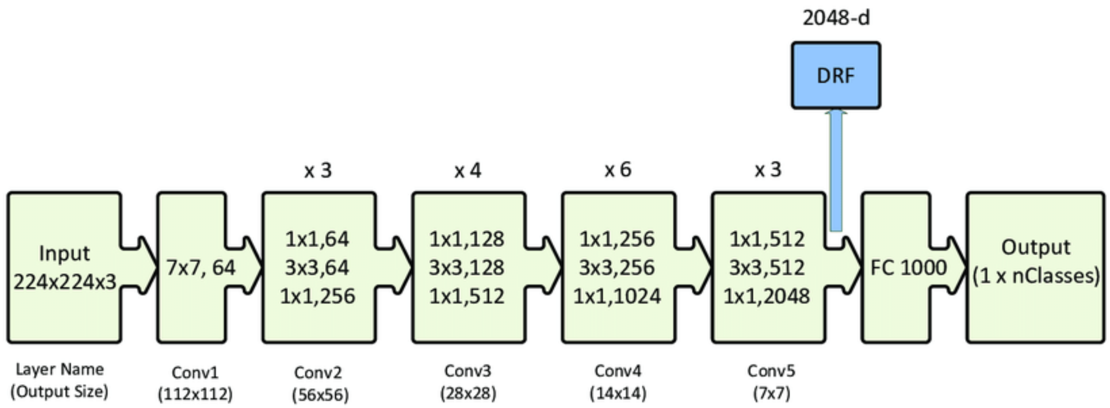

Fig 3.9.2: Architecture of ResNet50

ResNet50 is 50 layers deep Neural Net. One variation of the ResNet model, called ResNet50, consists of 48 Convolution layers, 1 MaxPool layer, and 1 Average Pool layer [24]. 3.8 x 109 Floating-point operations are possible. Following data management, the machine accepts data as input. Immediately after acquiring a picture, preprocessing begins. After pictures have been preprocessed, the ResNet50 model is applied to them. The image is then classified with the primary objective of extracting the feature. Then, the best model is utilized to predict the outcome. Fig 3.9.2 depicts the structure of ResNet50.

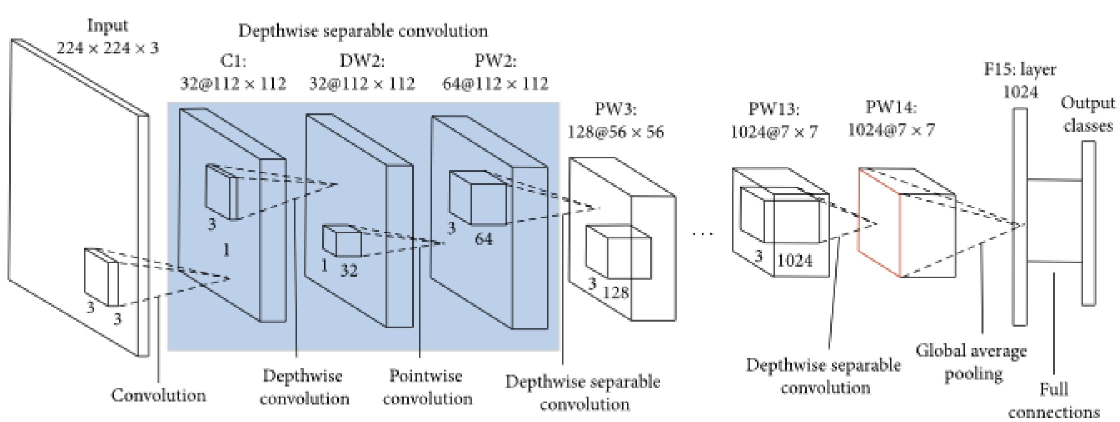

Fig 3.9.3: Architecture of MobileNetV2

The convolutional layers of **MobileNetV2's** deep neural network total 53 [25]. A version of the network that has been trained using the ImageNet database of over 1 million images is available for loading. The pretrained network can sort images into a thousand different categories based on the objects shown in the picture. With equal model sizes and computational costs, MobileNetV2 beats MobileNetV1 and ShuffleNet (1.5). MobileNetV2 (1.4) surpasses ShuffleNet (2) and NASNet with faster inference time because of its wider width multiplier. Fig 3.9.3 shows the architecture of MobileNetV2.

3.10 Training the Model

The primary objective of the training model was to retrain previously learned CNN models. VGG16, Resnet50, and MobileNetV2 are the pre-trained models deployed for this investigation. We utilized Google colab for programming because it provided the GPU required to effectively train the models. In the process of training, the data utilized by the models have undergone data augmentation and preprocessing. For the process of training models, it is necessary to determine the batch size, train steps, and epochs as shown below.

Batch Size:

The Batch Size parameter specifies how many independent data trains should be fed into the neural network in one go into the system in each successive train stage. This research used a batch size of 32 for both our datasets, which signifies that 32 pieces of information will be sent into the neural network at each stage.

Train Steps:

The total amount of data in the training set divided by the batch size yields the train stapes. We used two different datasets so the training steps will also differ according to their training sets. We know, Train Steps = Training Set / Batch Size

For KU-BdSL, Train Steps = 9600 / 32 = 300

For BDSL 49, Train Steps = 6923 / 32 = 216.34375 = 217

Epoch

Epoch is a unit of time that is used to measure the duration of a cycle of training. For KU-BdSl, 20 time periods were used for this analysis. What this means is that we perform 20 iterations of training and validation on the neural networks using the data. In terms of BDSL 49, we used 10 epochs for training our models.

3.11 Hyper Parameter Tuning

A neural network's parameters are most often the connection weights. These factors are learned in this context. As a result, these settings are fine-tuned by the algorithm (in conjunction with the incoming data). The hyper parameters are typically the learning rate, batch size, and the number of epochs. Hyper parameters determine the nature of the learning process. Performance may be tweaked using hyper settings.

Table 3.10.1: Hyper Parameter Tuning

No	Name of Hyper Parameter	Tuning
1	Epochs	10, 20
2	Batch Size	32
3	Image Size	224*224
4	Training Steps	217, 300
5	Rescale	1./255
6	Class_mode	Categorical
7	Optimizer	Adam
8	Model	Sequential
9	Dropout	0.5, 0.6
10	Train, Test	80%, 20%

3.12 Parameter

After projecting the model summary, all parameters are clearly visible. We utilized two distinct datasets hence the parameter differs between them. In KU-BdSL, the total parameter of VGG16 is about 21,145,182, the trainable parameter = 6,430,494, and the non-trainable parameter = 14,714,688. In ResNet50 the total parameter = 49,285,790, the trainable parameter = 25,698,078, and the non-trainable parameter = 23,587,712. Finally, if we look at MobileNetV2, the total parameter = 4,139,614, the trainable parameter = 1,881,630, and the non-trainable parameter = 2,257,984.

In BDSL 49, the total parameter of VGG16 is about 21,146,724, the trainable parameter = 6,432,036, and the non-trainable parameter = 14,714,688. But in ResNet50 the total parameter = 49,287,332, the trainable parameter = 25,699,620, and the non-trainable parameter = 23,587,712. At last, we have MobileNetV2 with a total parameter of 4,515,940, the trainable parameter = 2,257,956, and the non-trainable parameter = 2,257,984.

CHAPTER 4
EXPERIMENTAL RESULTS AND DISCUSSION

4.1 Introduction

In this chapter, we provide the results of our research. Our findings will be presented in two formats due to the fact that we used two different datasets. Table 4.1.1 shows the accuracy we gain from different models using KU-BdSL dataset. On the other hand, Table 4.1.2 shows the accuracy of the models, where we used BDSL 49 dataset.

Table 4.1.1: Accuracy of Models in KU-BdSL

Serial No.	Model	Accuracy
1	VGG16	98%
2	ResNet50	97%
3	MobileNetV2	95%

Table 4.1.2: Accuracy of Models in BDSL 49

Serial No.	Model	Accuracy
1	VGG16	90%
2	ResNet50	85%
3	MobileNetV2	67%

4.2 Experimental Result

This is a crucial aspect of the training process is comparing the performance of different models. Because of this comparison, we can exactly how our model is performing.

Model Evaluation Using KU-BdSL

This evaluation is recorded by using KU-BdSL dataset, which contained the single-handed image set of Bengali Sign Language.

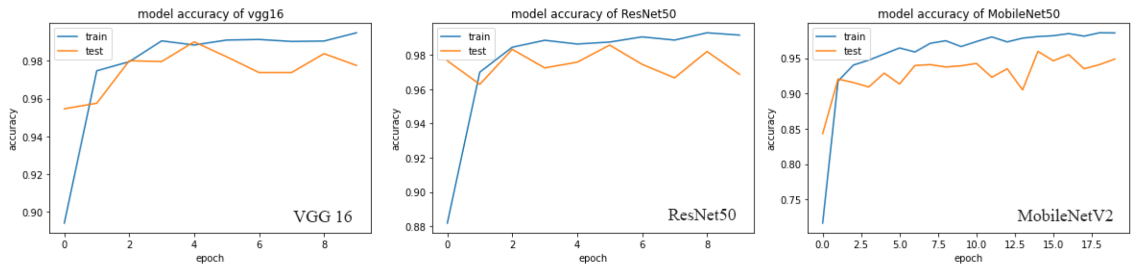

Fig 4.2.1: Model Accuracy (KU-BdSL)

When we look at Fig 4.2.1, we can observe that both the training accuracy and the validation accuracy are rapidly increasing as each epoch passes. It is clear from the graphs that VGG16 is performing significantly better than the other models.

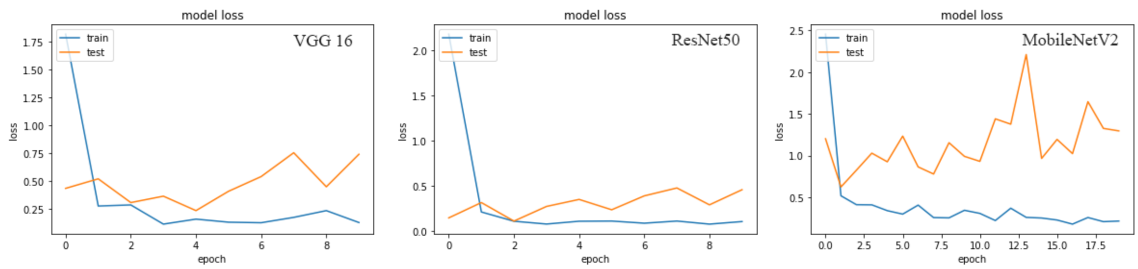

Fig 4.2.2: Model Loss (KU-BdSL)

Training loss and validation loss both decrease with time in Fig 4.2.2, with the exception of MobileNetV2. Once again, VGG16 had superior results than the others.

Model Evaluation Using BDSL 49

BDSL 49 is a dual-handed image set of Bengali Sign Language. And this evaluation bais sed on BDSL 49.

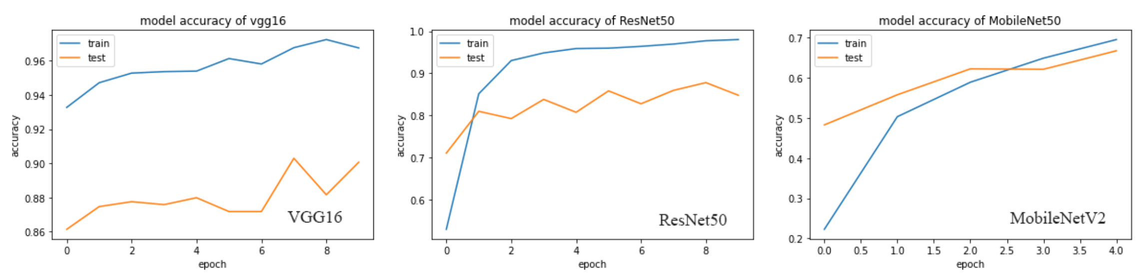

Fig 4.2.3: Model Accuracy (BDSL 49)

From Fig 4.2.3, we can see that both the training accuracy and the validation accuracy have been steadily improving over all of the epochs. VGG16 had a performance that was marginally superior to that of the others.

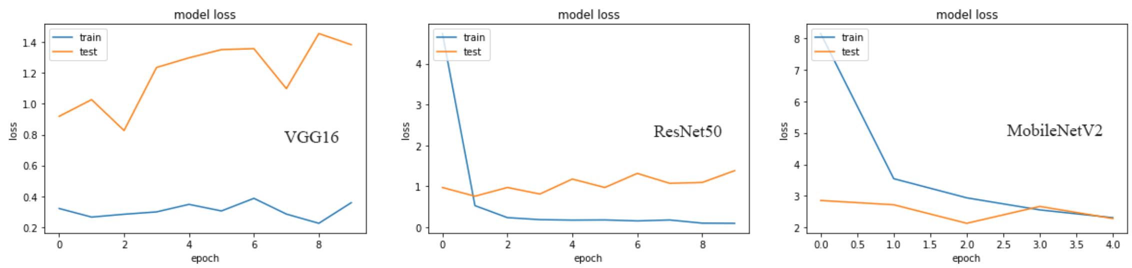

Fig 4.2.4: Model Loss (BDSL 49)

VGG16 was able to minimize the losses more effectively than other models, as evidenced by the training loss and validation loss graph in Figure 4.2.4.

4.3 Result Analysis & Discussion

The performance of various classifier models may be evaluated with the use of the confusion matrix. It does this by compiling the results of the predictions made about a categorization challenge. The percentage of right predictions (see True Positive, True Negative, False Positive, False Negative), counts, and criteria for categorizing each result are all displayed. The confusion matrix summarizes all of the outcomes and shows you where your classification model could get confused while making predictions. The accuracy of the classification model can be enhanced by determining the frequency and kind of model errors.

Confusion Matrix Plot using KU-BdSL

We plotted two confusion matrices for each model. One is for determining the predictions of the test set and the other is to visualize the predictions of the evaluation set.

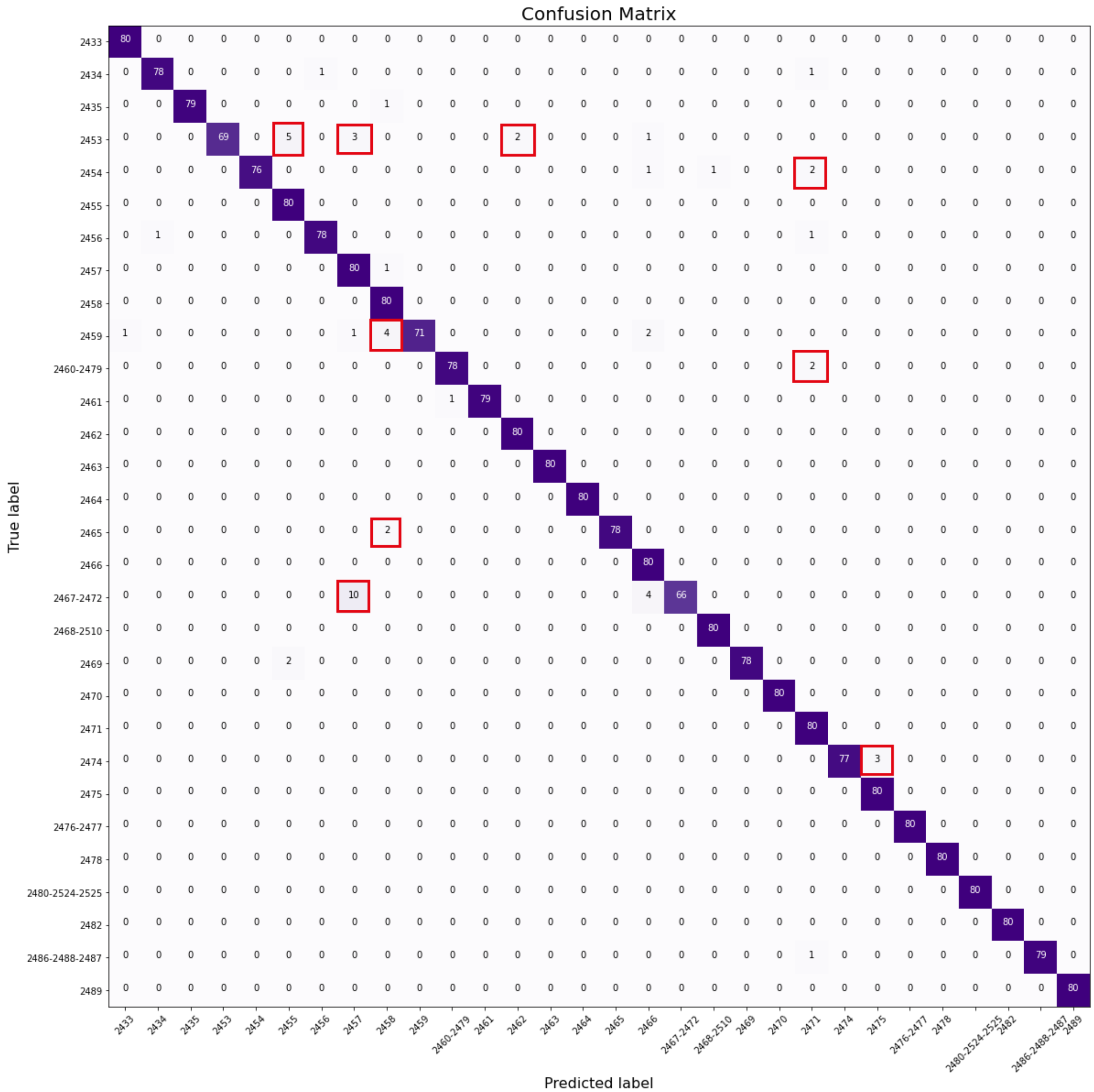

Fig 4.3.1: Confusion Matrix of VGG16 (KU-BdSL)

After analyzing the figures of the confusion matrix VGG16 recognized the images into different categories well. In class 2467-2472 the performance is not that good. In that class, there is a miss calculation and more than 10 images were predicted incorrectly. Some other errors as well.

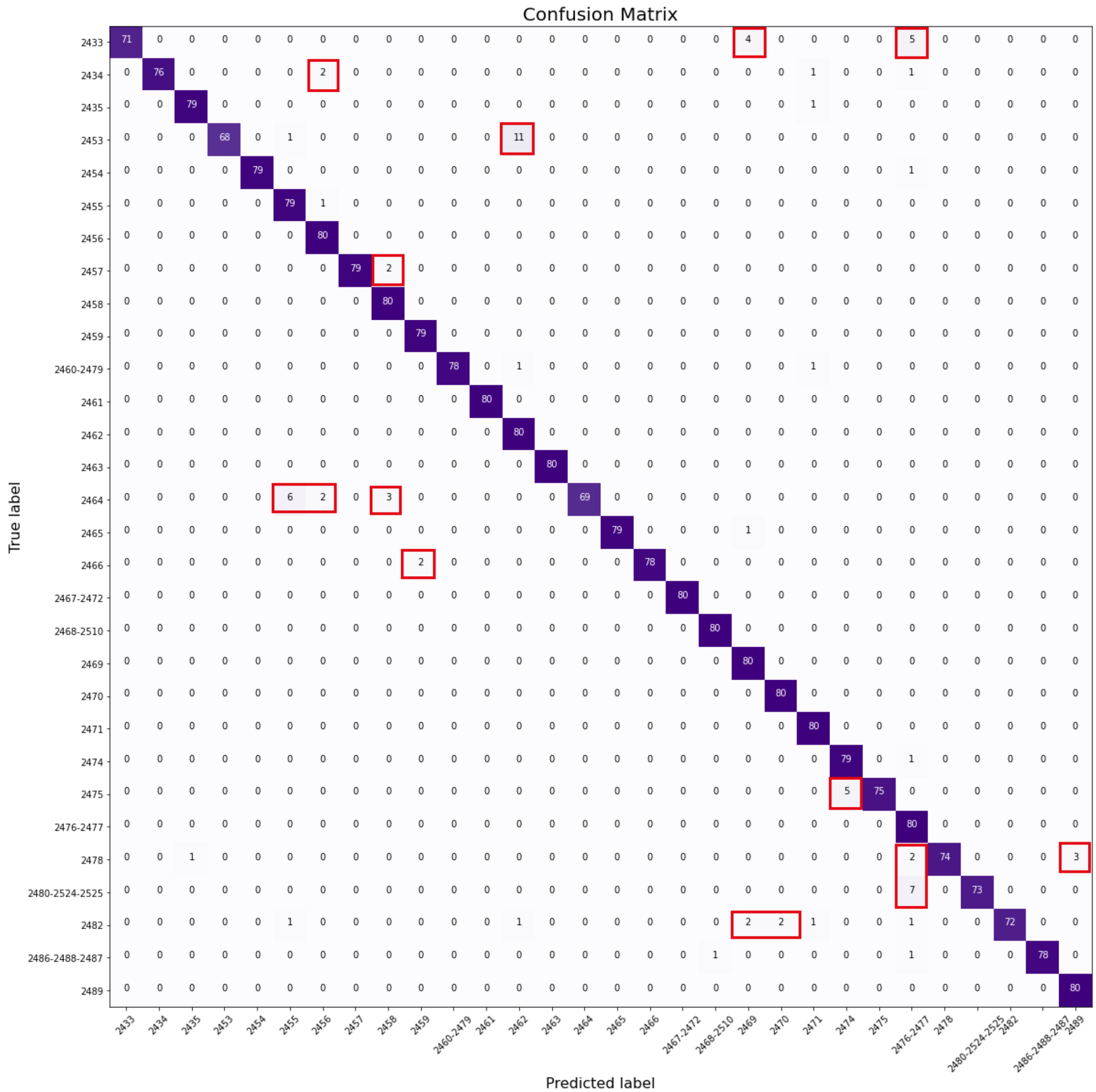

Fig 4.3.2: Confusion Matrix of ResNet50 (KU-BdSL)

After conducting research on the figures of the confusion matrix, ResNet50 was able to correctly classify the photographs into their respective groups. The overall result in class 2453 is not very impressive. In that particular group, there is an error in the formula, and more than eleven pictures have been mistakenly predicted.

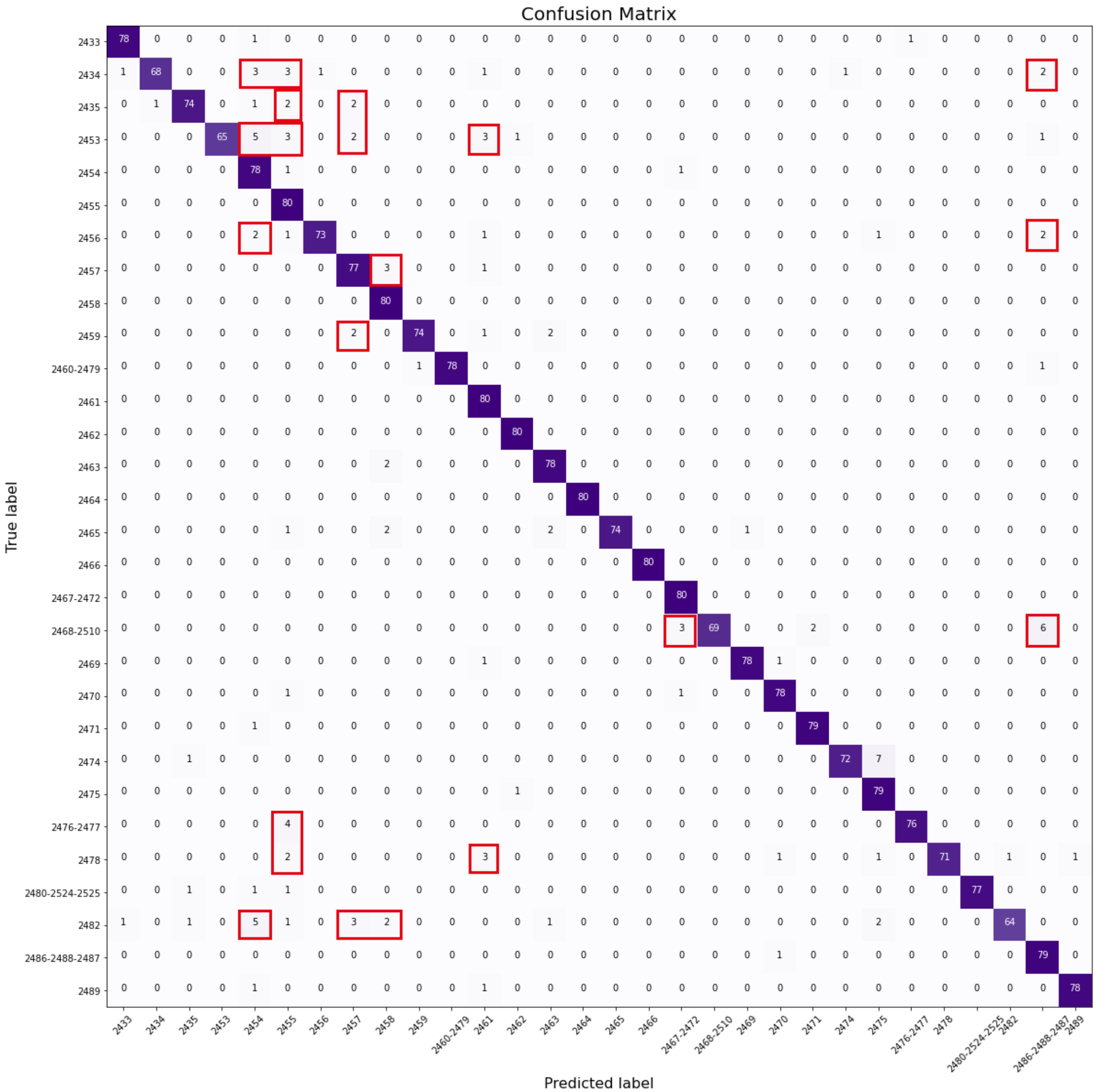

Fig 4.3.3: Confusion Matrix of MobileNetV2 (KU-BdSL)

The errors or wrong detections were highlighted with red markings. After analyzing the figures of confusion matrices above, we can say that VGG16 is recognized better than other models. MobileNetV2 performed well in visualizing the test set but it struggled a lot in terms of the evaluation set.

Confusion Matrix Plot using BDSL 49

Here we also plotted two confusion matrices for each model. One is for determining the predictions of the test set and the other is to visualize the predictions of the evaluation set. The confusion matrices of BDSL 49 are larger than the confusion matrices of KU-BdSL. Because we took 36 alphabets from the BDSL 49.

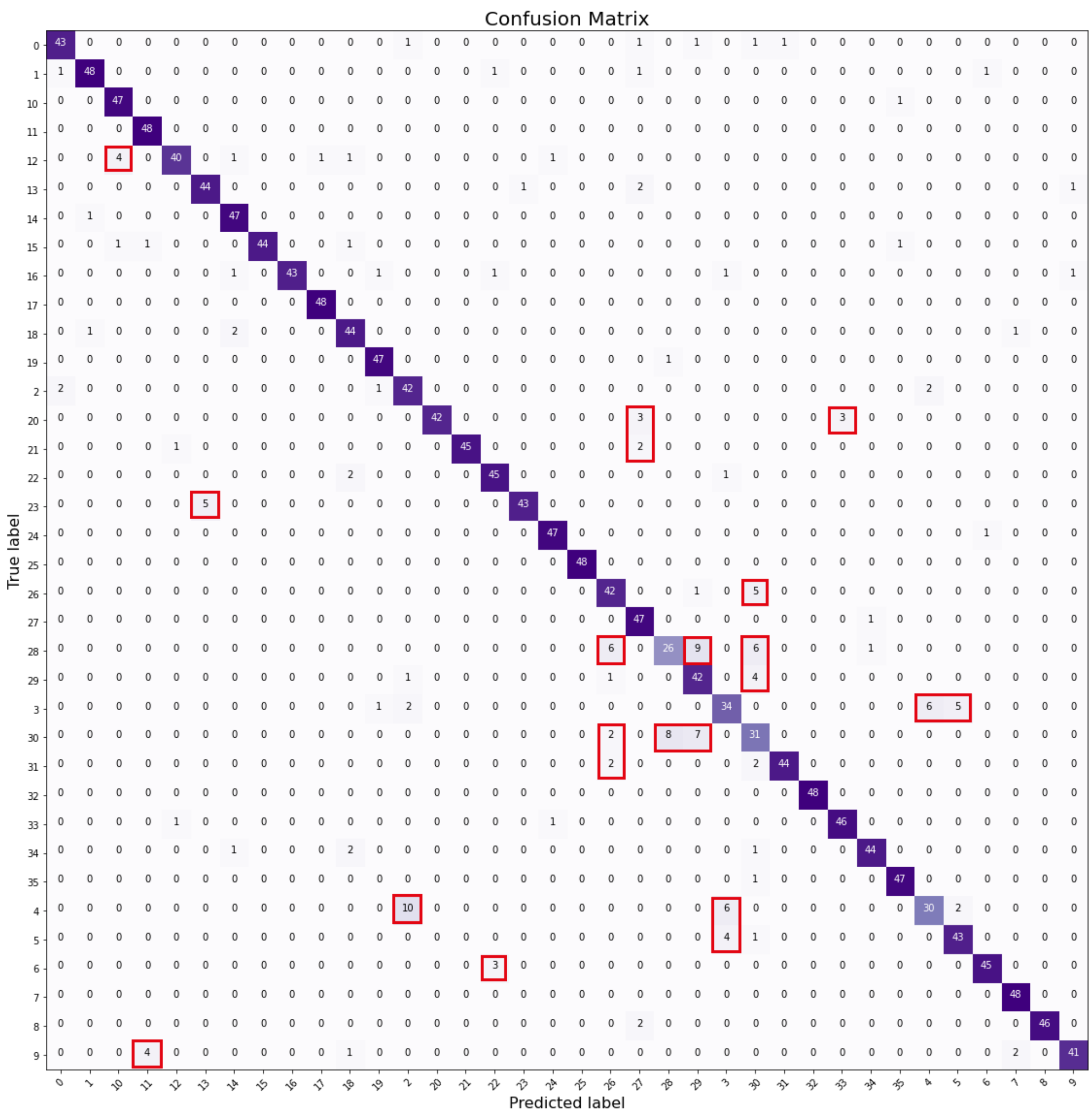

Fig 4.3.4: Confusion Matrix of VGG16 (BDSL 49)

VGG16 was able to classify photos into their appropriate groups after examining the confusion matrix's numbers. Class 4 does not have very strong results. Around 20 photos were inaccurately predicted in that class, therefore there is a miss calculation there.

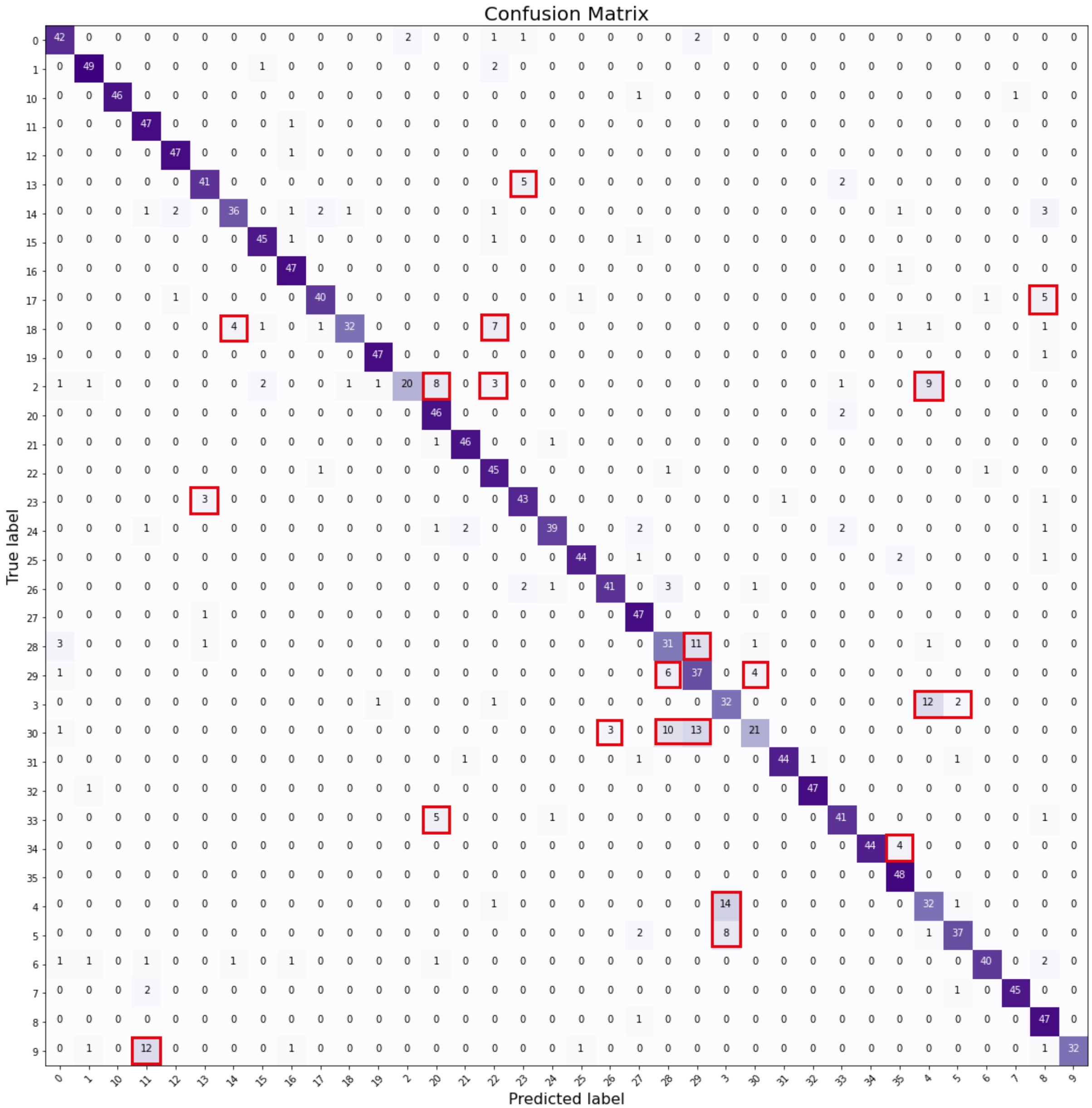

Fig 4.3.5: Confusion Matrix of ResNet50 (BDSL 49)

After conducting research on the figures of the confusion matrix, ResNet50 was able to correctly classify the photographs into their respective groups. The number of incorrect predictions is higher than the value of VGG 16.

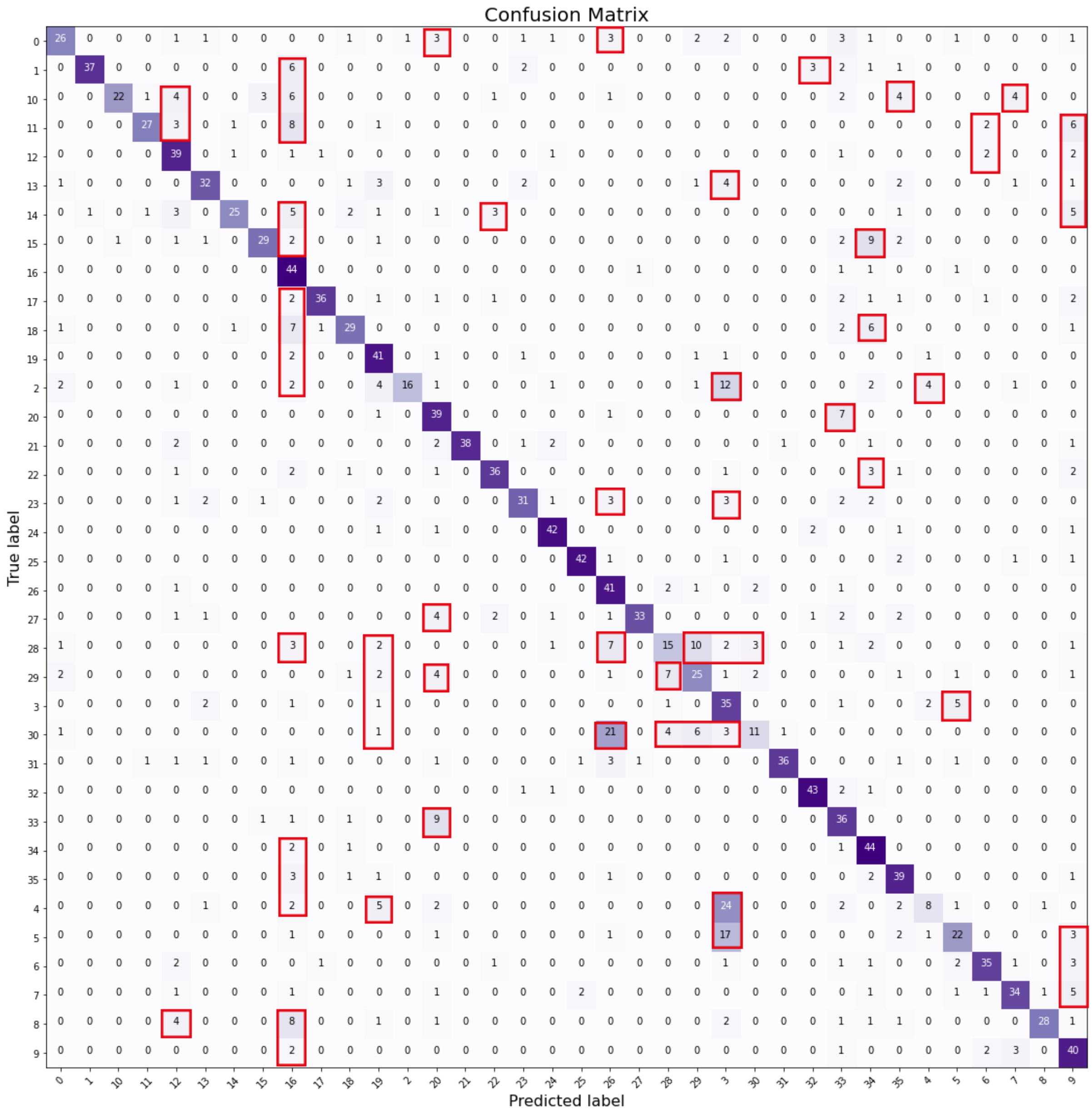

Fig 4.3.6: Confusion Matrix of MobileNetV2 (BDSL 49)

According to these confusion matrices, the errors are visualized with red markings. We can evaluate that, in terms of BDSL 49 VGG16 outperformed all the other models. It recognizes with much more sophistication than others. MobileNetV2 performed so poorly on the prediction of dual-handed BdSL.

Classification Report

Classification reports measure an algorithm's predictions. Unstructured or structured data of any given dataset can be used in machine learning for classification into labels, targets, categories, etc. in a predictive modeling process that starts with the class prediction of the given data points and then approximates the input variables mapping function to discrete variables as the output to recognize the classification of the new data points in space and class. Which forecasts came true? True Positives, False Positives, True Negatives, and False Negatives predict categorization report metrics. Before we dive into the classification report, we need to understand Precision, Recall, and F1-Score.

Precision

Precision refers to the proportion of positive classes accurately predicted out of all positive classes.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}} \tag{1}$$

Recall

Recall indicates the proportion of positive classifications for which our predictions were accurate. The more the values, the higher the quality of the model.

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}} \tag{2}$$

F1-Score

The F1-score is a useful tool for measuring both recall and precision at the same time. In place of the arithmetic mean, it makes use of the harmonic mean.

$$\text{F1Score} = \frac{2(\text{Recall} * \text{Precision})}{\text{Recall} + \text{Precision}} \tag{3}$$

Classification Report using KU-BdSL

Table 4.3.1: Classification Report of VGG16 (KU-BdSL)

Labels	Alphabets	Precision	Recall	F1-score	Support
2433	(ঁ)	0.99	1.00	0.99	80
2434	(ং)	0.99	0.97	0.98	80
2435	(ঃ)	1.00	0.99	0.99	80
2453	ক	1.00	0.86	0.93	80
2454	খ	1.00	0.95	0.97	80
2455	গ	0.92	1.00	0.96	80
2456	ঘ	0.99	0.97	0.98	80
2457	ঙ	0.85	0.99	0.91	81
2458	চ	0.91	1.00	0.95	80
2459	ছ	1.00	0.90	0.95	79
24560-2479	জ / য	0.99	0.97	0.98	80
2461	ঝ	1.00	0.99	0.99	80
2462	ঞ	0.98	1.00	0.99	80
2463	ট	1.00	1.00	1.00	80
2464	ঠ	1.00	1.00	1.00	80
2465	ড	1.00	0.97	0.99	80
2466	দ	0.91	1.00	0.95	80
2467-2472	ন / ণ	1.00	0.82	0.90	80
2468-2510	ত/ৎ	0.99	1.00	0.99	80
2469	থ	1.00	0.97	0.99	80
2470	দ	1.00	1.00	1.00	80
2471	ধ	0.92	1.00	0.96	80
2474	প	1.00	0.96	0.98	80
2475	ফ	0.96	1.00	0.98	80
2476-2477	ব/ভ	1.00	1.00	1.00	80
2478	ম	1.00	1.00	1.00	80
2480-2524-2525	র/ড়/ঢ়	1.00	1.00	1.00	80
2482	ল	1.00	1.00	1.00	80
2486-2488-2487	শ/স/ষ	1.00	0.99	0.99	80
2489	হ	1.00	1.00	1.00	80
Accuracy				0.98	2400
Macro avg		0.98	0.98	0.98	2400
Weighted avg		0.98	0.98	0.98	2400

Table 4.3.2: Classification Report of ResNet50 (KU-BdSL)

Labels	Alphabets	Precision	Recall	F1-score	Support
2433	(ঁ)	1.00	0.89	0.94	80
2434	(ং)	1.00	0.95	0.97	80
2435	(ঃ)	0.99	0.99	0.99	80
2453	ক	1.00	0.85	0.92	80
2454	খ	1.00	0.99	0.99	80
2455	গ	0.91	0.99	0.95	80
2456	ঘ	0.94	1.00	0.97	80
2457	ঙ	1.00	0.98	0.99	81
2458	চ	0.94	1.00	0.97	80
2459	ছ	0.98	1.00	0.99	79
24560-2479	জ / য	1.00	0.97	0.99	80
2461	ঝ	1.00	1.00	1.00	80
2462	ঞ	0.86	1.00	0.92	80
2463	ট	1.00	1.00	1.00	80
2464	ঠ	1.00	0.86	0.93	80
2465	ড	1.00	0.99	0.99	80
2466	দ	1.00	0.97	0.99	80
2467-2472	ন / ণ	1.00	1.00	1.00	80
2468-2510	ত/ৎ	0.99	1.00	0.99	80
2469	থ	0.92	1.00	0.96	80
2470	দ	0.98	1.00	0.99	80
2471	ধ	0.95	1.00	0.98	80
2474	প	0.94	0.99	0.96	80
2475	ফ	1.00	0.94	0.97	80
2476-2477	ব/ভ	0.81	1.00	0.89	80
2478	ম	1.00	0.93	0.96	80
2480-2524-2525	র/ড়/ঢ়	1.00	0.91	0.95	80
2482	ল	1.00	0.90	0.95	80
2486-2488-2487	শ/স/ষ	1.00	0.97	0.99	80
2489	হ	0.96	1.00	0.98	80
Accuracy				0.97	2400
Macro avg		0.97	0.97	0.97	2400
Weighted avg		0.97	0.97	0.97	2400

Table 4.3.3: Classification Report of MobileNetV2 (KU-BdSL)

Labels	Alphabets	Precision	Recall	F1-score	Support
2433	(ঁ)	0.97	0.97	0.97	80
2434	(ং)	0.99	0.85	0.91	80
2435	(ঃ)	0.96	0.93	0.94	80
2453	ক	1.00	0.81	0.90	80
2454	খ	0.80	0.97	0.88	80
2455	গ	0.80	1.00	0.89	80
2456	ঘ	0.99	0.91	0.95	80
2457	ঙ	0.90	0.95	0.92	81
2458	চ	0.90	1.00	0.95	80
2459	ছ	0.99	0.94	0.96	79
24560-2479	জ / য	1.00	0.97	0.99	80
2461	ঝ	0.87	1.00	0.93	80
2462	ঞ	0.98	1.00	0.99	80
2463	ট	0.94	0.97	0.96	80
2464	ঠ	1.00	1.00	1.00	80
2465	ড	1.00	0.93	0.96	80
2466	দ	1.00	1.00	1.00	80
2467-2472	ন / ণ	0.94	1.00	0.97	80
2468-2510	ত/ৎ	1.00	0.86	0.93	80
2469	থ	0.99	0.97	0.98	80
2470	দ	0.96	0.97	0.97	80
2471	ধ	0.98	0.99	0.98	80
2474	প	0.99	0.90	0.94	80
2475	ফ	0.88	0.99	0.93	80
2476-2477	ব/ভ	0.99	0.95	0.97	80
2478	ম	1.00	0.89	0.94	80
2480-2524-2525	র/ড়/ঢ়	1.00	0.96	0.98	80
2482	ল	0.98	0.80	0.88	80
2486-2488-2487	শ/স/ষ	0.87	0.99	0.92	80
2489	হ	0.99	0.97	0.98	80
Accuracy				0.95	2400
Macro avg		0.95	0.95	0.95	2400
Weighted avg		0.95	0.95	0.95	2400

Classification Report using BDSL 49

This classification report is prepared by using the dataset BDSL 49

Table 4.3.4: Classification Report of VGG16 (BDSL 49)

Labels	**Alphabets**	**Precision**	**Recall**	**F1-score**	**Support**
0	অ	0.93	0.90	0.91	48
1	আ	0.96	0.92	0.94	52
2	ই	0.90	0.98	0.94	48
3	উ	0.91	1.00	0.95	48
4	এ	0.95	0.83	0.89	48
5	ও	0.90	0.92	0.91	48
6	ক	0.90	0.98	0.94	48
7	খ	1.00	0.92	0.96	48
8	গ	1.00	0.90	0.95	48
9	ঘ	0.98	1.00	0.99	48
10	চ	0.86	0.92	0.89	48
11	ছ	0.94	0.98	0.96	48
12	জ	0.75	0.89	0.82	47
13	ঝ	1.00	0.88	0.93	48
14	ট	1.00	0.94	0.97	48
15	ঠ	0.90	0.94	0.92	48
16	ড	0.98	0.90	0.93	48
17	ঢ	0.96	0.98	0.97	48
18	ত	1.00	1.00	1.00	48
19	থ	0.79	0.88	0.83	48
20	দ	0.81	0.98	0.89	48
21	ধ	0.74	0.54	0.63	48
22	ন	0.70	0.88	0.78	48
23	প	0.74	0.71	0.72	48
24	ফ	0.60	0.65	0.62	48
25	ব	0.98	0.92	0.95	48
26	ভ	1.00	1.00	1.00	48
27	ম	0.94	0.96	0.95	48
28	য়	0.96	0.92	0.94	48
29	র	0.96	0.98	0.97	48
30	ল	0.79	0.62	0.70	48
31	স	0.86	0.90	0.88	48
32	হ	0.96	0.94	0.95	48
33	ড়	0.94	1.00	0.97	48
34	(ং)	1.00	0.96	0.98	48
35	(ঃ)	0.95	0.85	0.90	48
Accuracy				0.90	1731
Macro avg		0.90	0.90	0.90	1731
Weighted avg		0.90	0.90	0.90	1731

Table 4.3.5: Classification Report of ResNet50 (BDSL 49)

Labels	Alphabets	Precision	Recall	F1-score	Support
0	অ	0.86	0.88	0.87	48
1	আ	0.92	0.94	0.93	52
2	ই	1.00	0.96	0.98	48
3	উ	0.73	0.98	0.84	48
4	এ	0.94	0.98	0.96	48
5	ও	0.89	0.85	0.87	48
6	ক	0.88	0.75	0.81	48
7	খ	0.92	0.94	0.93	48
8	গ	0.89	0.98	0.93	48
9	ঘ	0.91	0.83	0.87	48
10	চ	0.94	0.67	0.78	48
11	ছ	0.96	0.98	0.97	48
12	জ	0.91	0.43	0.58	47
13	ঝ	0.74	0.96	0.84	48
14	ট	0.94	0.96	0.95	48
15	ঠ	0.73	0.94	0.82	48
16	ড	0.84	0.90	0.87	48
17	ঢ	0.93	0.81	0.87	48
18	ত	0.96	0.92	0.94	48
19	থ	0.93	0.85	0.89	48
20	দ	0.84	0.98	0.90	48
21	ধ	0.61	0.65	0.63	48
22	ন	0.59	0.77	0.67	48
23	প	0.59	0.67	0.63	48
24	ফ	0.78	0.44	0.56	48
25	ব	0.98	0.92	0.95	48
26	ভ	0.98	0.98	0.98	48
27	ম	0.85	0.85	0.85	48
28	য়	1.00	0.92	0.96	48
29	র	0.84	1.00	0.91	48
30	ল	0.57	0.67	0.62	48
31	স	0.88	0.77	0.82	48
32	হ	0.95	0.83	0.89	48
33	ড়	0.98	0.94	0.96	48
34	(ং)	0.73	0.98	0.84	48
35	(ঃ)	1.00	0.67	0.80	48
Accuracy				0.85	1731
Macro avg		0.86	0.85	0.85	1731
Weighted avg		0.86	0.85	0.85	1731

Table 4.3.6: Classification Report of MobileNetV2 (BDSL 49)

Labels	**Alphabets**	**Precision**	**Recall**	**F1-score**	**Support**
0	অ	0.76	0.54	0.63	48
1	আ	0.97	0.71	0.82	52
2	ই	0.96	0.46	0.62	48
3	উ	0.90	0.56	0.69	48
4	এ	0.59	0.81	0.68	48
5	ও	0.78	0.67	0.72	48
6	ক	0.89	0.52	0.66	48
7	খ	0.85	0.60	0.71	48
8	গ	0.39	0.92	0.55	48
9	ঘ	0.92	0.75	0.83	48
10	চ	0.76	0.60	0.67	48
11	ছ	0.59	0.85	0.70	48
12	জ	0.94	0.34	0.50	47
13	ঝ	0.53	0.81	0.64	48
14	ট	1.00	0.79	0.88	48
15	ঠ	0.82	0.75	0.78	48
16	ড	0.79	0.65	0.71	48
17	ঢ	0.82	0.88	0.85	48
18	ত	0.93	0.88	0.90	48
19	থ	0.48	0.85	0.62	48
20	দ	0.94	0.69	0.80	48
21	ধ	0.52	0.31	0.39	48
22	ন	0.53	0.52	0.53	48
23	প	0.32	0.73	0.45	48
24	ফ	0.61	0.23	0.33	48
25	ব	0.95	0.75	0.84	48
26	ভ	0.88	0.90	0.89	48
27	ম	0.49	0.75	0.60	48
28	য়	0.56	0.92	0.69	48
29	র	0.62	0.81	0.70	48
30	ল	0.50	0.17	0.25	48
31	স	0.63	0.46	0.53	48
32	হ	0.81	0.73	0.77	48
33	ড়	0.76	0.71	0.73	48
34	(ং)	0.93	0.58	0.72	48
35	(ঃ)	0.51	0.83	0.63	48
Accuracy				0.67	1731
Macro avg		0.67	0.67	0.67	1731
Weighted avg		0.67	0.67	0.67	1731

CHAPTER 5

IMPACT ON SOCIETY, ENVIRONMENT, AND SUSTAINABILITY

5.1 Impact on Society

Our ability to communicate with others in our own languages is something we frequently take for granted. Many people who are deaf are unable to benefit from even these most fundamental encounters due to the significant challenges associated with using sign language. There are more than 70 million deaf persons in the globe who rely on sign languages to communicate. By learning and using sign language, they have access to educational and career possibilities, as well as basic social services, allowing them to fully engage in society. Bangladeshi Sign Language (BdSL) is the native language of about 2.6 million deaf individuals in Bangladesh. Our study will help everyone to understand Bengali Sign Language more thoroughly and will help us to create better interaction with deaf and mute people in our country.

5.2 Impact on the Environment

The United Nations Convention on the Rights of Persons with Disabilities mandates that countries accept, enable, and encourage the use of sign languages so that people with disabilities can enjoy their rights on an equal footing with others. We all use various signs while talking to one another. However, sign language is the sole means of communication available to those who are deaf or hard of hearing. Because our research focuses on establishing Bengali Sign Language's legitimacy, it will pave the way for a community in which everyone has the same chance to create a better environment. We will be able to build an environment in which everyone may live in peace, harmony, and mutual respect.

5.3 Ethical Aspects

Research ethics refers to a set of guidelines that should be used to make decisions on how to conduct studies. When conducting a study, students, scientists, and researchers have a responsibility to maintain a particular level of professionalism. Trust in scientific discoveries, human rights, and dignity, and collaboration between scientists and the general public all benefit from a focus on ethics in research. Informed consent and the safety of study participants may be guaranteed if researchers adhere to these standards. We were able to collect data from legitimate sources which gave us a boost of research validity and maintain our study integrity.

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Summary of the Study

Convolutional neural networks were used in this study to do a comparative analysis between single-handed and dual-handed Bangladeshi Sign Language Recognition. This brief summary of our study will give an overview of this work.

Step 1

- Literature Review and Comparative analysis
- Data Collection and Processing
- Prepare the Data for Training

Step 2

- Selection of the Pre-Trained Models
- Compile and Fit the Models
- Training the Data using the Models

Step 3

- Evaluation of the Models
- Room for Improvement
- Set the Parameters

Final Step

- Determine the Predicted Test Images and Accuracy
- Tuning the Model Operators
- Determine the Final Accuracy

6.2 Conclusion

Ours is a study of comparative analysis of single and dual-handed Bengali Sign Language recognition. That's why we choose two different datasets. KU-BdSL was selected for single-handed Bengali Sign Language and BDSL 49 was selected for dual-handed Bengali Sign Language. We selected three pre-trained models from CNN, VGG16, ResNet50, and MobileNetV2. These models were applied in both of the datasets. Among them, VGG16 performed better than other models in both of our datasets. In KU-BdSL, we got 98% accuracy using VGG16, and in BDSL 49, we got 90% accuracy using the same models. The fact that CNN is one of the machine-learning algorithms that are most successful for picture classification is the driving force behind our decision to use it.

6.3 Implication for Further Study

In this experiment, we have shown recognition of both single and dual-handed Bengali Sign Language using two distinct datasets. This comparative analysis of both of them has opened the door for the future. In the future, hopefully, we will be able to work with depth-image or 3D imaging sets. This will help us to create an environment for real-time implementation as an android application or a web application where we can translate sign language.

Reference:

[1] "Deafness in Bangladesh," wikipedia.org. <<https://en.wikipedia.org/wiki/Deafness_in_Bangladesh>> last accessed on 15-11-2022 at 5:33 PM

[2] "International Day of Sign Languages," un.org. <<https://www.un.org/en/observances/sign-languages day>> last accessed on 15-11-2022 at 5:33 PM.

[3] "Bangladesh Sign language Day," dpiap.org. <<http://www.dpiap.org/national/article.php?countryid=017&id=0000029&country=Bangladesh#:~:text=Almost%202.6%20million%20deaf%20people,express%20and%20share%20with%20others>>. last accessed on 15-11-2022 at 5:33 PM.

[4] Jaid Jim, Abdullah Al; Rafi, Ibrahim; Akon, Md. Zahid; Nahid, Abdullah-Al (2021), "KU-BdSL: Khulna University Bengali Sign Language dataset", Mendeley Data, V1, doi: 10.17632/scpvm2nbkm.1

[5] Sizan Khan, Saqib; Hasib, Ayman; Eva, Jannatul Ferdous; Rahman, Rashik; Murad, Hasan; Islam, Md Rajibul; Hussein, Molla Rashied (2022), "BDSL 49: A Comprehensive Dataset of Bengali Sign Language", Mendeley Data, V5, doi: 10.17632/k5yk4j8z8s.5

[6] Rayeed, S. M., Sidratul Tamzida Tuba, Hasan Mahmud, Md. Saddam Hossain Mukta, Md. Mumtahin Habib Ullah Mazumder and Md. Kamrul Hasan. "Bdsl47: A Complete Open-Access Depth-Based Bangla Sign Alphabet and Digit Dataset." *SSRN Electronic Journal* (2022): n. pag.

[7] Md. Sanzidul Islam, Sadia Sultana Sharmin Mousumi, Nazmul A. Jessan, Nazmul A. Jessan, and Sayed Akhter Hossain, "Ishara-Lipi: The First Complete Multipurpose Open Access Dataset of Isolated Characters for Bangla Sign Language," Engineering Applications of Artificial Intelligence 76 (2018) 202–213, 8 September 2018.

[8] M.A Hossen, Arun Govindaiah, Sadia Sultana, and Alauddin Bhuiyan, "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), 25-29 June 2018.

[9] Joyeeta Singha and Karen Das, "Indian Sign Language Recognition Using Eigen Value Weighted Euclidean Distance Based Classification Technique," (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 4, No. 2, 2013.

[10] Oishee Bintey Hoque, Mohammad Imrul Jubair, Md. Saiful Islam, Al-Farabi Akash, and Alvin Sachie Paulson, "Real Time Bangladeshi Sign Language Detection using Faster R-CNN," International Conference on Innovation in Engineering and Technology (ICIET), 27-29 December, 2018.

[11] Shruti Chavan, Xinrui Yu, and Jafar Saniie, "Convolutional Neural Network Hand Gesture Recognition for American Sign Language," 2021 IEEE International Conference on Electro Information Technology (EIT), 14-15 May 2021.

[12] Angur M. Jarman, Samiul Arshad, Nashid Alam, and 4Mohammed J. Islam, "An Automated Bengali Sign Language Recognition System Based on Fingertip Finder Algorithm," International Journal of Electronics & Informatics, Vol.4, No.1, July 2015.

[13] Ravikiran J, Kavi Mahesh, Suhas Mahishi, Dheeraj R, Sudheender S and Nitin V Pujari, "Finger Detection for Sign Language Recognition," Proceedings of the International Multi Conference of Engineers and Computer Scientists Hong Kong, Vol I IMECS 2009, March 18 - 20, 2009.

[14] Mayand Kumar, Piyush Gupta, Rahul Kumar Jha, Aman Bhatia, Khushi Jha and Bickey Kumar Shah, "SIGN LANGUAGE ALPHABET RECOGNITION USING CONVOLUTION NEURAL NETWORK," Proceedings of the Fifth International Conference on Intelligent Computing and Control Systems (ICICCS 2021), 06-08 May 2021.

[15] Farhad Yasir, P.W.C. Prasad, Abeer Alsadoon, A. Elchouemi, and Sasikumaran Sreedharan, "Bangla Sign Language Recognition using Convolutional Neural Network," 2017 International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), 06-07 July 2017.

[16] Syed Tauhid Ahmed and M. A. H. Akhand, "Bangladeshi Sign Language Recognition using Fingertip Position," 2016 International Conference on Medical Engineering, Health Informatics and Technology (MediTec), 17-18 December 2016.

[17] A. Sharmila Konwar, B. Sagarika Borah and C. Dr.T.Tuithung, "A. Sharmila Konwar, B. Sagarika Borah, C. Dr.T.Tuithung," International Conference on Communication and Signal Processing, April 3-5, 2014, India.

[18] Wenjin Tao A, Ming C. Leu a and Zhaozheng Yin, "American Sign Language alphabet recognition using Convolutional Neural Networks with multiview augmentation and inference fusion," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987.

[19] Cao Dong, Ming C. Leu and Zhaozheng Yin, "Cao Dong, Ming C. Leu, and Zhaozheng Yin," 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 07-12 June 2015.

[20] Hasan Mahmud, Md. Kamrul Hasan, Abdullah-Al-Tariq, Md. Hasanul Kabir, and M. A. Mottalib, "Hindawi Advances in Human-Computer Interaction," IEEE Transl. J. Magn. Japan, Volume 2018, Article ID 1069823, 13 pages.

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

[22] Medium, available at <<https://medium.com/mlearning-ai/convolutional-neural-networks-cnns-1ed0f1f8e176>>, last accessed on 05-10-2022 at 5:31 PM.

[23] Research on unbalanced training samples based on SMOTE algorithm - Scientific Figure on ResearchGate, available at << https://www.researchgate.net/figure/Structure-of-VGGNet-16_fig2_335577870>>, last accessed on 05-10-2022 at 5:32 PM.

[24] Automatic Hierarchical Classification of Kelps Using Deep Residual Features - Scientific Figure on ResearchGate. Available from: <<https://www.researchgate.net/figure/ResNet-50-architecture-26-shown-with-the-residual-units-the-size-of-the-filters-and_fig1_338603223 >>, last accessed on 05-10-2022 at 5:33 PM.

[25] Medium, available at << https://medium.com/analytics-vidhya/image-classification-with-mobilenet-cc6fbb2cd470>>, last accessed on 05-10-2022 at 5:31 PM.

Final Report

ORIGINALITY REPORT

22%	19%	12%	14%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	6%
2	Submitted to Daffodil International University Student Paper	2%
3	www.researchgate.net Internet Source	1%
4	www.mdpi.com Internet Source	1%
5	A. Sharmila Konwar, B. Sagarika Borah, C. T. Tuithung. "An American Sign Language detection system using HSV color model and edge detection", 2014 International Conference on Communication and Signal Processing, 2014 Publication	1%
6	M.A Hossen, Arun Govindaiah, Sadia Sultana, Alauddin Bhuiyan. "Bengali Sign Language Recognition Using Deep Convolutional Neural Network", 2018 Joint 7th International Conference on Informatics, Electronics &	1%