BANGLA HANDWRITTEN SINGLE, NUMERAL, VOWEL MODIFIER AND COMPOUND CHARACTERS RECOGNITION USING CNN

BY

MEHADI HASSAN SOVON
ID: 181-15-10712
AND

SYED RAIHANUZZAMAN
ID: 181-15-10568
AND

SADIA JAMAN
ID: 181-15-10710

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

Supervised By
Md. Sazzadur Ahamed
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised By
Md. Zahid Hasan
Associate Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2021

APPROVAL

This Project/internship titled **"Bangla Handwritten Single, Numeral, Vowel Modifier and Compound Characters Recognition Using CNN"**, submitted by Mehadi hassan sovon, ID No: 181-15-10712 and Syed Raihanuzzaman, ID No: 181-15-10568 and Sadia jaman, ID No: 181-15-10710 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 5 December, 2022.

BOARD OF EXAMINERS

Chairman

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

Naznin Sultana
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner

Raja Tariqul Hasan Tusher
Senior Lecturer
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

External Examiner

Dr. Dewan Md. Farid
Professor
Department of Computer Science and Engineering
United International University

DECLARATION

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

Supervised by:

____

Md. Sazzadur Ahamed
Senior Lecturer
Department of CSE
Daffodil International University

Co-Supervised by:

____

Md. Zahid Hasan
Associate Professor
Department of CSE
Daffodil International University

Submitted by:

____

Mehadi Hassan Sovon
ID: 181-15-10712
Department of CSE
Daffodil International University

Syed Raihanuzzaman

____

Syed Raihanuzzaman
ID: 181-15-10568
Department of CSE
Daffodil International University

Sadia

____

Sadia Jaman
ID: 181-15-10712
Department of CSE
Daffodil International University

ACKNOWLEDGEMENT

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

We really grateful and wish our profound our indebtedness to **Md. Sazzadur Ahamed, Senior Lecturer, Department of CSE** Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of "web application development" 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 stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Professor Dr. Touhid Bhuiyan, Head**, **Department of CSE**, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

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

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

ABSTRACT

The difficulty of handwritten character identification varies by language, owing to differences in shapes, lines, numbers, and size of characters. There are several studies for the identification of handwritten characters accessible for English in comparison with other significant languages like Bangla. In their recognition procedures, existing technologies use multiple techniques such as classification tools and feature extraction. CNN has recently been shown to be proficient in handwritten character recognition in English. A Handwritten Bangla character identification system based on CNN has been examined in this research. Using CNN, the suggested approach for feature, labeling, and normalizing the handwritten character of images, as well as categorizing different characters. It doesn't use a feature extraction approach like previous research in the field. This research used almost 4,50,000 unique handwritten characters in a variety of styles. The recommended model has been proved to have a high recognition accuracy level and outperforms some of the most widely used methods already in use.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

The world's population has extended to 7 billion across 195 countries. Almost every country has its own language and alphabet. Recognition of characters and alphabets is a broad area of research. Presently, recognition of characters has been accorded massive attention in interim printed forms and handwritten. Handwritten character recognition is more complex than printed forms because different types of people have different types of handwriting.

Bangla language holds the 5th rank in the world's spoken languages and is the mother tongue of Bangladesh. More than 200M people exert Bangla for speaking along with writing purposes. UNESCO declared February 21st as Mother Language Day Internationally to honor the martyrs who fought for their mother tongue in 1952 Bangladesh. Therefore, now is the right time to computerize the Bengali language. The Bangla language has 50 characters, including digits, vowel modifiers, and many compound letters. There are some similar shapes of characters and compound letters in the Bangla alphabet. That’s why achieving a better result is so difficult.

The handwriting recognition system consists of two main steps: extracting features from a dataset and classifying individual characters using learning tools. Convolutional Neural Networks (CNN) have grown popular in recent years for complicated visual identification due to its architecture and several research studies on utilizing deep CNN to detect handwritten numbers [1], characters [2] [3], and other complex optical recognition. Because of its unique characteristics, the convolutional neural network (CNN) has been deemed effective in recognizing handwritten characters.

1.2 Motivation

The motivation of this research is to identify Bangla's handwritten characters for application in various sectors. Nowadays in Bangladesh, most of the official work of the government is written in Bangla. Besides, organizations like the education sector, health sector, most of the private companies, automotive industry, and signature verification are often use Bangla. This paper provides a shift from the traditional technique of transcribing handwritten documents, reducing the number of people, extra expenses, and time it takes to complete the process. On the other hand, when going through the literature review, it is observed that most of the work is done by basic characters, compounds, modifiers, and digits separately. Hence, it motivates us to work on a CNN-based handwritten recognition system where we are going to use basic characters, compounds, modifiers, and digit combined in a paper and our own created model will be used. Hopefully, the system will eventually complement existing technologies that will help governments and companies to improve document handling efficiency, save storage space, and ensure document security.

1.3 Objectives

The purpose of the research is to identify the Bangla handwritten character and digits with the use of 189 classes consisting of 50 fundamental characters, 119 compound characters, 10 numerals, and 10 modifiers. Considering the whole region of the Bengali language group is a monumental effort that necessitates the use of effective and sophisticated deep learning algorithms as well as leading hardware to build a model in the shortest period possible. Compared to the previous era of state-of-the-art systems on the handwritten Bangla letter and digit identification, we believe it is essential that the model now provides the least predicted validation accuracy.

1.4 Report Layout

Here, we work on a CNN model for recognition of Handwritten character. This report has five parts as shown below:

Chapter 1: Introduction

We provide an overview of the research project, including its motivation and objectives.

Chapter 2: Background Study

This chapter goes through the literature review, research summary, and challenges.

Chapter 3: Research Methodology

This chapter explains the research process, data preprocessing, data collection, and implementation.

Chapter 4: Experimental Results and Discussion

This section graphically depicts the outcome and experimental analysis, and research results.

Chapter 5: Summary, Conclusion & Future Work

This part includes an overview of our findings and recommendations for further work.

CHAPTER 2

BACKGROUND STUDY

2.1 Literature Review

Many previous experiments for handwritten character identification in a different language, such as Hindi and English, have had significant success.

A few studies on the handwritten fundamental digit, character, and compound character recognition in Bangla are available. Some literature on the identification of Bangla characters has been published in recent years. The previous experiment in Bangla character recognition has concentrated mainly on the Bangla digit and character, with ten digits and fifty characters.

In recent years, some literature has documented the recognition of Bangla characters as " Using MLP and SVM classifier Basic Bangla and Compound handwritten character recognition [4]." The recommended models worked with their own datasets, which consisted of 250 participants, and achieved the recognition rates of MLP and SVM of around 79.25% and 80.51%, respectively.

Md. Mahbubar Rahman et al. [5] work on Bangla's 50 isolated characters, normalizing the dataset and using the CNN approach for classification. The dataset consisted of 20,000 samples only, resized samples of each character in the 28X28 dimension. The founding rate of recognition is 85.96%.

Rumman Rashid Chowdhury et al. [6] worked on the BanglaLekha-Isolated dataset consisting of over 2,00,000 data which was expanded by data augmentation. After preprocessing, the selected data was 1,66,105, consisting of 50 characters, ten digits, and 24 compound letters. After implementing the CNN, the model achieved an accuracy of 95.25%. But the paper focused on only 50 characters. However, using digits and compound letters would have been more appreciable.

Bishwajit Purkayastha et al. [7] proposed a handwritten character recognition scheme using CNN and tested it on the BanglaLekha-Isolated dataset. The accuracy rate of numerals is 98.66%, vowels 94.99%, compound letters (20 classes) 91.60%, alphabets 91.23%, and the accuracy rate of Bengali characters is around 80 classes, 89.93%. However, adding more classes would have given more accuracy.

Chandrika Saha et al.[8] focused on BBCNet-15 for recognizing handwritten characters, and it's the pre-trained model of DCNN(Deep Convolutional Neural Network). Using the dataset of this scheme CMATERdb 3.1.2.Rate of the accuracy based on Bangla basic 50 characters 96.40%. However, the numerals and compound letters will be attached to it.

S M Azizul Hakim et al.[9] used a 9-layer sequential CNN model which is used to detect 60 Bangla handwritten characters, 17,645 Numerals data of 10 digits, 83,458 data of 50 basic characters are collected from BanglaLekha-Isolated dataset for the training validation and used as their own dataset for a test set where 25 students participate. The result of the existing dataset is 99.44%, and the test set is 95.16%, which is prepared.

AKM Shahariar Azad et al. [2] introduced the CNN model for identifying Handwriting Letters, including 50 fundamental Bangla characters. Experiments were conducted using three datasets: BanglaLekha-Isolated, ISI dataset, and CMATERdb. Using the BornoNet model, validation accuracy of CMATERdb is 98%, ISI dataset 96.81%, BanglaLekha-Isolated dataset 95.71%, and combined dataset 96.40%.

AKM Shahariar Azad et al. [3] works on the EkushNet model to identification Bangla handwritten 50 primary characters,10 modifiers,10 digits, and 52 compound characters used mostly. For validate and cross-validate Ekush dataset and CMATERdb dataset were used, respectively. The accuracy for the Ekush dataset is 97.73% and 95.01% for the CMATERdb dataset. However, the result would have been better if more prominent architecture could have been used.

Tandra Rani Das et al. [10] developed an extended CNN model to identification Bangla handwritten characters. The model has experimented with the “Bangla Lekha-Isolated” dataset consisting of 10 numerals, 39 consonants, and 11 vowels. The accuracy of Bangla numerals is 99.50%, vowels 93.18%, consonants 90.00%,combined classes 92.25%. However, using more classes and compound letters would have been great.

Md. Jahid Hasan et al. [11] came up with a Handwritten system that used both CNN, BiLSTM (Bidirectional long short-term memory). The model was tested on the CMATERdb 3.1.3.3 dataset consisting of 171 classes. The accuracy of the model is 98.50%. The model was only tested with compound letters. However, adding digits, simple characters, and modifiers would have been more promising.

2.2 Research Summary

Several research sectors, such as machine learning, data mining, deep learning, etc., identify Bangla handwritten characters. We addressed various kinds of research papers in the section of the literature review. Some of them use machine learning approaches like SVM classifier, gradient feature, and those who use deep learning approaches like CNN use less amount dataset. Also, most of them use pre-trained models. Thus research has been focused on creating own model and using a vast amount of datasets by combining two separate datasets.

2.3 Challenges

There will be difficulties in research work. But facing the problems and overcoming the situation by appropriately managing these problems can lead to good research work. As we are working on handwritten characters, we first try to collect data physically. But due to the Covid-19 situation, it was pretty challenging to get a proper dataset. Although we collected some data, their resolution was not enough because of not having quality mobile devices. As a result, we used existing datasets. Moreover, we didn't work on a pre-trained model, so it was quite challenging to create our model. By working on a vast dataset, pc configurations play a vital role, which was also problematic because of our low-quality PC configurations and also ran across several issues throughout the training as we utilized a vast dataset, and training took considerable time. But, in the end, we were able to overcome those obstacles and reach great accuracy. With test images, our CNN models also have performed admirably.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Methodological Overview:

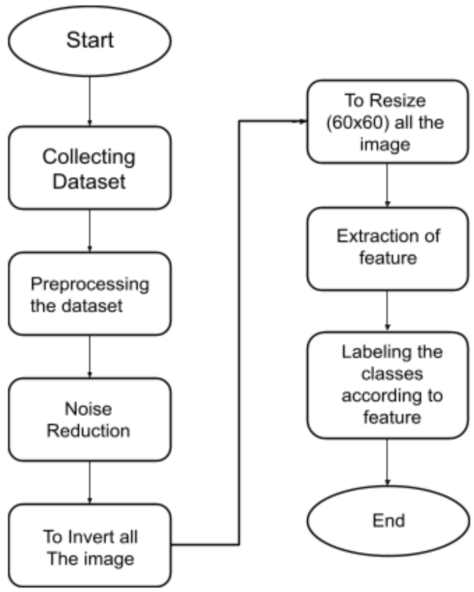


Figure 3.1: Methodological overview of propose method.

3.2 Dataset Collection:

For this research paper, we used two different types of datasets, called the "EkushNet"[12] and "BanglaLekha Isolated Dataset"[13]. After combining datasets, 50 characters consisting of 11 vowels and 39 consonants shown as Figure 3.2 and 3.3, ten numerals shown as Figure 3.4, ten modifiers shown as Figure 3.6, and 119 compound letters shown as Figure 3.5 were collected, modified, and preprocessed. Hence, the total number of data samples is around 450000, down from 500000, which is under 189 classes.

The Ekush dataset’s image resolutions are different because of character size and diverse people's writing styles, the dataset comprises a wide range of unique characters. Several of these character representations are shaped in a highly complicated way and are tightly connected. Bangla’s writing starts from left to right. The majority of Bangla characters feature an aclinic line in the top section, as can be seen. This line is known as Matra/Shirorekh[2]

When a consonant or a vowel following a consonant takes on a compound orthographic form, it is called a compound character[2]. Compound characters may be made up of two consonants or a consonant and a vowel.

Bangla allows for the compounding of multi-characters. In Bangla, the number of compound characters is 200. This study looks at recognizing such complex-shaped compound characters, using 119 prominent Bangla compound characters as examples.

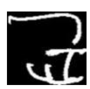
1	2	3	4	5	6	7	8	9	10
অ	আ	ই	ঈ	উ	ঊ	এ	ঐ	ও	ঔ
									

Figure 3.2: Bangla basic vowel character

1	2	3	4	5	6	7	8	9	10
ক	খ	গ	ঘ	ঙ	চ	ছ	জ	ঝ	ঞ
									

Figure 3.3: Bangla basic consonant character

0	1	2	3	4	5	6	7	8	9

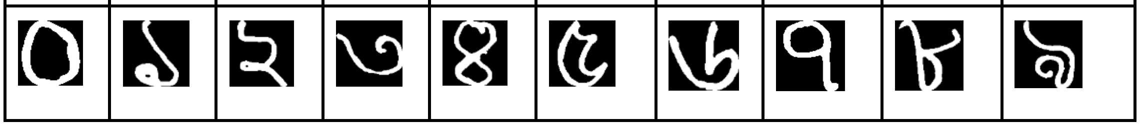


Figure 3.4: Bangla numerals

1	2	3	4	5	6	7	8	9	10
ন্ড	ড্ড	ষ্ঠ	ল্প	ষ্প	ন্দ	ন্ধ	ম্ম	ণ্ঠ	প্ত

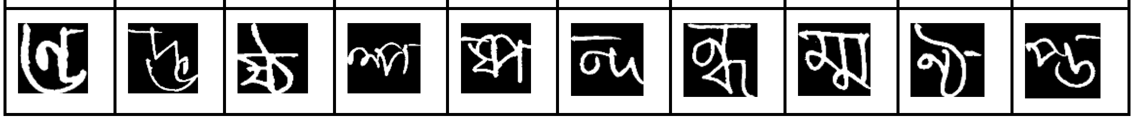


Figure 3.5: Bangla compound character

1	2	3	4	5	6	7	8	9	10
◌া	◌ি	◌ী	◌ু	◌ূ	◌ে	◌ো	◌ৈ	◌ৃ	◌ৌ

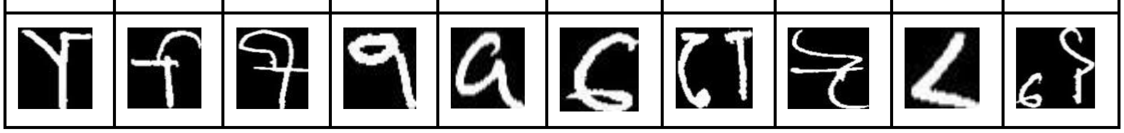


Figure 3.6: Bangla vowel modifiers

3.3 Pre-processing of Dataset:

Preprocessing transforms random pictures into a consistent shape or form that may be fed into classifiers. In deep learning, data preparation plays a vital role. Data is ubiquitous, but the issue is that it isn't processed. The proposed model uses "BanglaLekha Isolated Dataset" and "EkushNet". The character image of the EkushNet dataset is black, and the backdrop is white. First of all, the character of all images is switched to white, and the backdrop is switched to black (Figure 3.7), representing that the black pixels are 0, and the white pixels are 255. Both dataset images' size, height, and widthwise are not the same. As a result, the datasets were resized to 60 x 60 pixels (Figure 3.8).

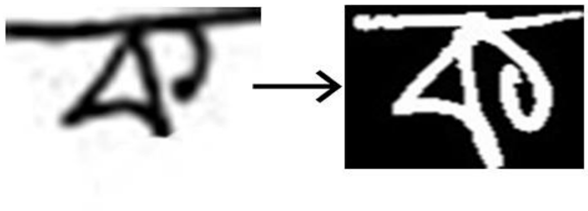


Figure 3.7: Inverted image

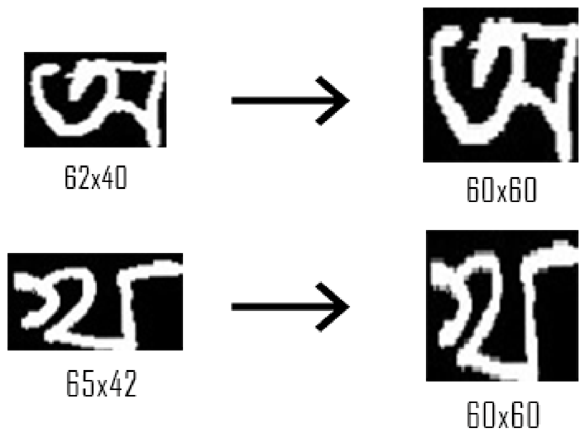


Figure 3.8: Resize Image

3.4 Feature & Label Extraction:

The entire data set has been feature extracted After resizing. Here no specific algorithm is used for feature extraction. Then, according to feature extraction, find the image's label. After that, all features and labels are stored in .npy file.

3.5 CNN Architecture:

To classify Bangla handwritten character identification were used the CNN approach with multilayer in this proposed method. Convolution, Max pooling layer, fully connected dense layer, and regularization techniques such as batch normalization[14] and dropout[15] were employed in this model.

A softmax layer is the ultimate output layer.

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{x_j}} \qquad (1)$$

the output layer provides a distribution of probabilities across a predetermined set of categories. the network recommends the category which has the most significant probability. The first convolutional layer collects information from the picture directly, allowing them to be utilized in a subsequent discrimination task. this layer searches the picture for receptive fields. The characteristics would be included in these fields. The characteristics are then transmitted to the relu activation mechanism after scanning for receptive fields.

$$Relu(x) = \max(0, x) \qquad (2)$$

3.5.1 Architecture for character:

This proposed method firstly used the Convolution 2D layer with a filter size of 32 where kernel size is 3x3 and the layer used the 'relu'() activation with the same padding. Then those layers are connected with a max-pooling layer. After that, for regularization 20% dropout layers were used.

Then the output of the previous layer goes through another Convolution 2D layer with a filter size of 64 with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts.

Layer7, the final convolution 2D layer, takes input from the last layer. It has 128 filters with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts.

Then, after these nine procedures, the result is flattened inside an array and sent through a fully connected dense layer of 128 filters, and for regularization, 20% dropout was used.

The previous layer's output will then be connected with a fully connected dense layer of 50 nodes which is the final layer for the model using the Softmax() activation.

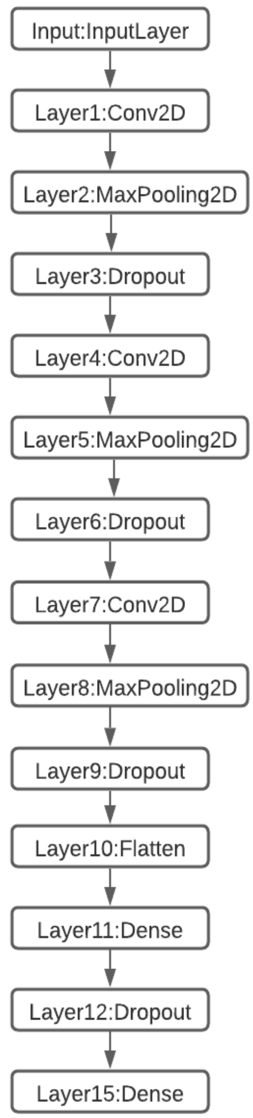


Figure 3.9: Architecture of Character

```
model.summary()
```

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_6 (Conv2D)            (None, 58, 58, 32)        896
_________________________________________________________________
max_pooling2d_6 (MaxPooling2 (None, 29, 29, 32)        0
_________________________________________________________________
dropout_8 (Dropout)          (None, 29, 29, 32)        0
_________________________________________________________________
conv2d_7 (Conv2D)            (None, 27, 27, 64)        18496
_________________________________________________________________
max_pooling2d_7 (MaxPooling2 (None, 13, 13, 64)        0
_________________________________________________________________
dropout_9 (Dropout)          (None, 13, 13, 64)        0
_________________________________________________________________
conv2d_8 (Conv2D)            (None, 11, 11, 128)       73856
_________________________________________________________________
max_pooling2d_8 (MaxPooling2 (None, 5, 5, 128)         0
_________________________________________________________________
dropout_10 (Dropout)         (None, 5, 5, 128)         0
_________________________________________________________________
flatten_2 (Flatten)          (None, 3200)              0
_________________________________________________________________
dense_4 (Dense)              (None, 128)               409728
_________________________________________________________________
dropout_11 (Dropout)         (None, 128)               0
_________________________________________________________________
dense_5 (Dense)              (None, 50)                6450
=================================================================
Total params: 509,426
Trainable params: 509,426
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.10: Model summary of Character

3.5.2 Architecture for digit:

Initially, this method used the Convolution 2D layer with a filter size of 16 where kernel size is 3x3 and the layer used the 'relu'() activation with the same padding. Then those layers are connected with a max-pooling layer. After that, for regularization 20% dropout layers were used. Then the output of the previous layer goes through another Convolution 2D layer with a filter size of 32 with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts.

Layer7, the final convolution 2D layer, takes input from the last layer. It has 64 filters with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts.

Then, After these nine procedures, the result is flattened inside an array and sent through a fully connected dense layer of 16 filters, and for regularization, 20% dropout was used. Then the output of the previous layer went through a second fully connected dense layer of 32. The previous layer's output will then be connected with a fully connected dense layer of 10 nodes which is the final layer for the model using the Softmax () activation.

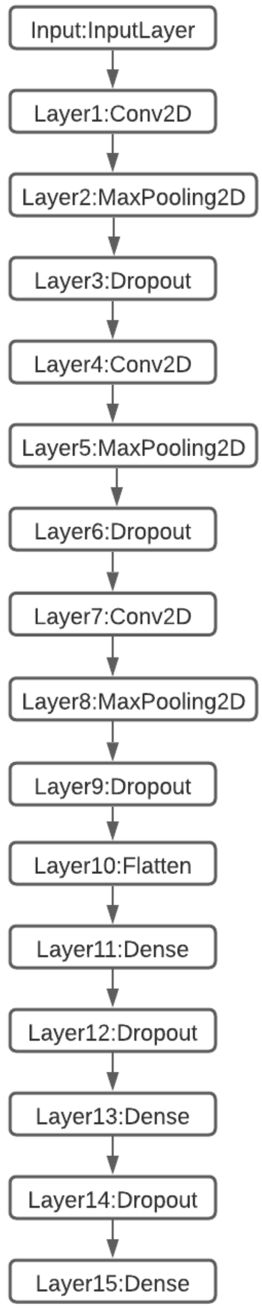


Figure 3.11: Architecture of digit

```
model.summary()
```

```
Model: "sequential_3"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_9 (Conv2D)            (None, 58, 58, 16)        448
_________________________________________________________________
max_pooling2d_9 (MaxPooling2 (None, 29, 29, 16)        0
_________________________________________________________________
dropout_12 (Dropout)         (None, 29, 29, 16)        0
_________________________________________________________________
conv2d_10 (Conv2D)           (None, 27, 27, 32)        4640
_________________________________________________________________
max_pooling2d_10 (MaxPooling (None, 13, 13, 32)        0
_________________________________________________________________
dropout_13 (Dropout)         (None, 13, 13, 32)        0
_________________________________________________________________
conv2d_11 (Conv2D)           (None, 11, 11, 64)        18496
_________________________________________________________________
max_pooling2d_11 (MaxPooling (None, 5, 5, 64)          0
_________________________________________________________________
dropout_14 (Dropout)         (None, 5, 5, 64)          0
_________________________________________________________________
flatten_3 (Flatten)          (None, 1600)              0
_________________________________________________________________
dense_6 (Dense)              (None, 16)                25616
_________________________________________________________________
dropout_15 (Dropout)         (None, 16)                0
_________________________________________________________________
dense_7 (Dense)              (None, 32)                544
_________________________________________________________________
dropout_16 (Dropout)         (None, 32)                0
_________________________________________________________________
dense_8 (Dense)              (None, 10)                330
=================================================================
Total params: 50,074
Trainable params: 50,074
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.12: Model summery of digit

3.5.3 Architecture for modifiers:

Our proposed method firstly used the Convolution 2D layer with a filter size of 16 where kernel size is 3x3 and the layer used the 'relu'() activation with the same padding. Then those layers are connected with a max-pooling layer. After that, for regularization 20% dropout layers were used. Then the output of the previous layer goes through another Convolution 2D layer with a filter size of 32 with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts. Layer 7, the final convolution 2D layer, takes input from the last layer. It has 64 filters with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts.Then, After these nine procedures, the result is flattened inside an array and sent through a fully connected dense layer of 16 filters, and for regularization, 20% dropout was used. The previous layer's output will then be connected with a fully connected dense layer of 10 nodes which is the final layer for the model using the Softmax() activation.

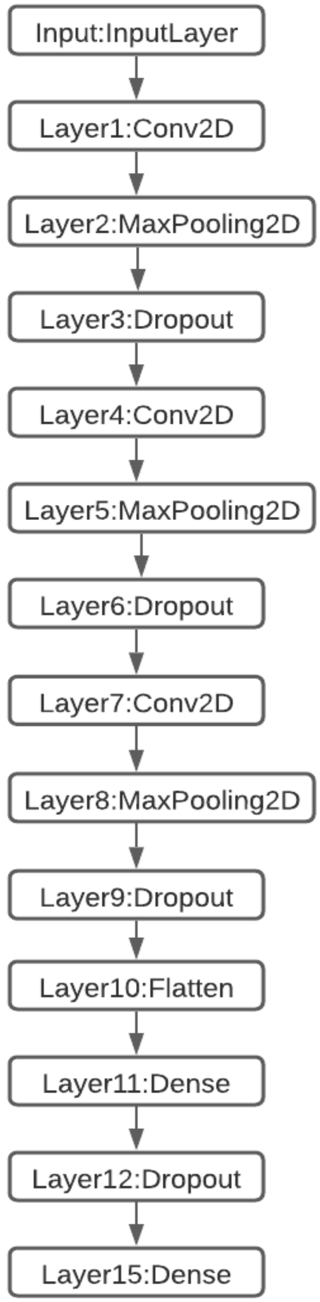


Figure 3.13: Architecture of modifier

```
model.summary()
```

```
Model: "sequential_4"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_12 (Conv2D)           (None, 58, 58, 16)        448
_________________________________________________________________
max_pooling2d_12 (MaxPooling (None, 29, 29, 16)        0
_________________________________________________________________
dropout_17 (Dropout)         (None, 29, 29, 16)        0
_________________________________________________________________
conv2d_13 (Conv2D)           (None, 27, 27, 32)        4640
_________________________________________________________________
max_pooling2d_13 (MaxPooling (None, 13, 13, 32)        0
_________________________________________________________________
dropout_18 (Dropout)         (None, 13, 13, 32)        0
_________________________________________________________________
conv2d_14 (Conv2D)           (None, 11, 11, 64)        18496
_________________________________________________________________
max_pooling2d_14 (MaxPooling (None, 5, 5, 64)          0
_________________________________________________________________
dropout_19 (Dropout)         (None, 5, 5, 64)          0
_________________________________________________________________
flatten_4 (Flatten)          (None, 1600)              0
_________________________________________________________________
dense_9 (Dense)              (None, 16)                25616
_________________________________________________________________
dropout_20 (Dropout)         (None, 16)                0
_________________________________________________________________
dense_10 (Dense)             (None, 10)                170
=================================================================
Total params: 49,370
Trainable params: 49,370
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.14: Model summery of modifier

3.5.4 Architecture of compound character:

This proposed method firstly used the Convolution 2D layer with a filter size of 16 where kernel size is 3x3 and the layer used the 'relu'() activation with the same padding. Then those layers are connected with a max-pooling layer. After that, for regularization, 20% dropout layers were used then the output of the previous layer goes through another Convolution 2D layer with a filter size of 32 with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts. Layer 7, the final convolution 2D layer, takes input from the last layer. It has 64 filters with a 3x3 kernel size, where the activation is 'relu,' connected with a max-pooling layer with 20% dropouts.

Then, after these nine procedures, the result is flattened inside an array and sent through a fully connected dense layer of 64 filters, and for regularization, 20% dropout was used. The previous layer's output will then be connected with a fully connected dense layer of 119 nodes which is the final layer for the model using the Softmax() activation.

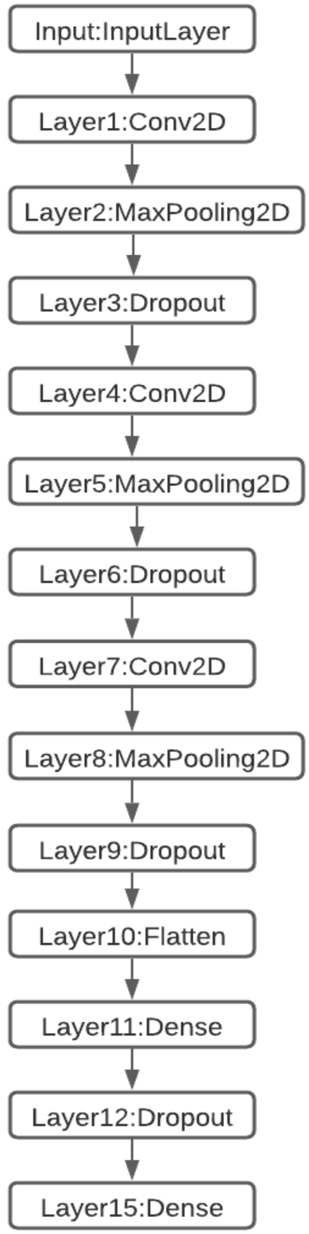


Figure 3.15: Architecture of compound character

```
model.summary()
```

```
Model: "sequential_5"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_15 (Conv2D)           (None, 58, 58, 16)        448
_________________________________________________________________
max_pooling2d_15 (MaxPooling (None, 29, 29, 16)        0
_________________________________________________________________
dropout_21 (Dropout)         (None, 29, 29, 16)        0
_________________________________________________________________
conv2d_16 (Conv2D)           (None, 27, 27, 32)        4640
_________________________________________________________________
max_pooling2d_16 (MaxPooling (None, 13, 13, 32)        0
_________________________________________________________________
dropout_22 (Dropout)         (None, 13, 13, 32)        0
_________________________________________________________________
conv2d_17 (Conv2D)           (None, 11, 11, 64)        18496
_________________________________________________________________
max_pooling2d_17 (MaxPooling (None, 5, 5, 64)          0
_________________________________________________________________
dropout_23 (Dropout)         (None, 5, 5, 64)          0
_________________________________________________________________
flatten_5 (Flatten)          (None, 1600)              0
_________________________________________________________________
dense_11 (Dense)             (None, 64)                102464
_________________________________________________________________
dropout_24 (Dropout)         (None, 64)                0
_________________________________________________________________
dense_12 (Dense)             (None, 119)               7735
=================================================================
Total params: 133,783
Trainable params: 133,783
Non-trainable params: 0
_________________________________________________________________
```

Figure 3.16: Model summery of compound

3.6 Optimizer:

CNN techniques benefit from optimization algorithms since they help to reduce error. The Adam optimizer [16] was applied in the proposed model. The Adam optimization technique may be used to iteratively update network weights in the training phase. Adam is a version of the stochastic gradient descent technique that has been extended. It is commonly utilized in deep learning-based research because of its superior performance.

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt[1]{\hat{v}} + \varepsilon} \widehat{mt} \qquad (3)$$

We implemented the categorical cross-entropy function to determine the error for the optimization procedure. According to the latest studies, cross-entropy outperforms other functions such as classification error and mean squared error [17].

$$L_i = -\sum_j t_{i,\ j} \log \left(p_{i,\ j}\right) \qquad (4)$$

CHAPTER 4

EXPERIMENTAL RESULTS AND DISCUSSION

On our dataset, the proposed model was trained and verified. And provide positive outcomes on a train, test, and validation set.

4.1 Train, Test and Validation split

Train test and validation splits were developed to measure the model's performance. All features and labels have been linked together.

The character dataset has 101368 data images and this dataset is split into four parts and each part has been contained 25342 characters, in this proposed method applied 75% of data for training purposes and 25% data for test purposes.

Same as the character dataset, the digit dataset has 19697 data images. The dataset is split into two parts and each part contains 9848 digits; it means 50% of image data. And applied 50% data for the train and 50% for the test.

The modifier dataset has 30550 data images. Like other datasets, the dataset is split into two parts and each part contains 15275 modifier images that mean 50% of the data. And here the model applied 50% data for the train and 50% for the test.

The compound character dataset has 303888 data images and this dataset is split into eight parts and each part contains 37986 characters, in this proposed method applied 265902 images, which means 87.5% of data for training purposes and 12.5% data for testing purposes.

4.2. Model performance

For single character model was used 40 epochs. After that, it got a train accuracy of 95.71% and a test accuracy of 84.62%. Then for numerals 50 epoch were used. It got the train accuracy of 92.77% and a test accuracy of 94.00%. After 40 epochs modifiers got the train accuracy of 95.46% and a test accuracy 96.46%. Finally, for the compound letters the epoch 30 was used and it also got the train accuracy of 74.98% and a test accuracy of 77.60%.

Table 1: Test accuracy for different section of model

TYPE	CLASS	TEST ACCURACY	TEST LOSS
Single characters	50	84.62%	0.7554
Numerals	10	94%	0.2094
Modifiers	10	96.46%	0.1482
Compound letters	119	77.60%	0.7967

Table 2: Train accuracy for different section of model

TYPE	CLASS	TRAIN ACCURACY	TRAIN LOSS
Single characters	50	95.71%	0.1251
Numerals	10	92.77%	0.2207
Modifiers	10	95.46%	0.1299
Compound letters	119	74.98%	0.8208

Table 3: Validation accuracy for different section of model

TYPE	CLASS	VALIDATION ACCURACY	VALIDATION LOSS
Single characters	50	85.34%	0.7248
Numerals	10	96.24%	0.1355
Modifiers	10	96.27%	0.1406
Compound letters	119	77.55%	0.7825

CHAPTER 5

SUMMARY, CONCLUSION AND FUTURE WORK

5.1 Summary of the Study

To detect Bangla's handwritten characters, we applied deep learning in this study. This work is broken down into multiple sections such as topic selection, data collection, data preprocessing, execution of methodology, and Experiment evaluation. We use two existing datasets combinedly. Using google Colab and Jupyter Notebook, we preprocess our dataset by resizing all images, inverting image background, etc. We have fixed to use CNN after study on related research paper. One of the key advantages of CNN is that it is compatible with new technology. Our CNN model was trained by Jupyter. Without using a pre-trained model, our proposed model gets good accuracy. Improving the accuracy, we use different types of own created models in different sections of our work. We get our best result from our own model using the CNN method.

5.2 Conclusions and Future Work

This research aims to see how well CNN can classify Bangla handwritten characters, compound letters, digits, and up/lower signs. A convolutional neural network (CNN) can identify optical patterns directly away from pixel images with minimum preparation. As a result, the paper examines a CNN structure for handwritten pattern classification with only the raw feature. It also showed that utilizing a larger quantity of data with variance might aid the model's learning of the classes' attributes or qualities. The accuracy of our proposed work is average because it is done with our created model. We can utilize more deep layers or pre-trained models in the future.

APPENDICES

Abbreviation

CNN = Convolutional Neural Network

SVM = Support Vector Machine

MLP = Multi layer Perception

BiLSTM = Bidirectional long short-term memory

Appendix: Observations on the Research

When We started, we knew nothing about machine learning, deep learning, or recognition algorithms. Our supervisor has been a wonderful person who is always willing to help. He was quite helpful and provided important advice from the start. We learnt a lot over the period of the research, including how to create a superior dataset, how to develop CNN models, or how to avoid overfitting. Ultimately, We learned about several deep learning and machine learning methods as an outcome of our study, and it has encouraged us to go further in the future.

REFERENCE

[1] Rabby, A. S. A., Abujar, S., Haque, S., & Hossain, S. A. (2019). Bangla handwritten digit recognition using convolutional neural network. In *Emerging Technologies in Data Mining and Information Security* (pp. 111-122). Springer, Singapore.

[2] Rabby, A. S. A., Haque, S., Islam, S., Abujar, S., & Hossain, S. A. (2018). Bornonet: Bangla handwritten characters recognition using convolutional neural network. *Procedia computer science*, *143*, 528-535.

[3] Rabby, A. S. A., Haque, S., Abujar, S., & Hossain, S. A. (2018). Ekushnet: using convolutional neural network for bangla handwritten recognition. *Procedia computer science*, *143*, 603-610.

[4] Das, N., Das, B., Sarkar, R., Basu, S., Kundu, M., & Nasipuri, M. (2010). Handwritten Bangla basic and compound character recognition using MLP and SVM classifier. *arXiv preprint arXiv:1002.4040*.

[5] Rahman, M. M., Akhand, M. A. H., Islam, S., Shill, P. C., & Rahman, M. H. (2015). Bangla handwritten character recognition using convolutional neural network. *International Journal of Image, Graphics and Signal Processing*, *7*(8), 42.

[6] Chowdhury, R. R., Hossain, M. S., ul Islam, R., Andersson, K., & Hossain, S. (2019, May). Bangla handwritten character recognition using convolutional neural network with data augmentation. In *2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)* (pp. 318-323). IEEE.

[7] Purkaystha, B., Datta, T., & Islam, M. S. (2017, December). Bengali handwritten character recognition using deep convolutional neural network. In *2017 20th International conference of computer and information technology (ICCIT)* (pp. 1-5). IEEE.

[8] Saha, C., Faisal, R. H., & Rahman, M. M. (2019, May). Bangla handwritten basic character recognition using deep convolutional neural network. In *2019 Joint 8th International Conference on Informatics, Electronics & Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)* (pp. 190-195). IEEE.

[9] Hakim, S. A. (2019, February). Handwritten bangla numeral and basic character recognition using deep convolutional neural network. In *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)* (pp. 1-6). IEEE.

[10] Das, T. R., Hasan, S., Jani, M. R., Tabassum, F., & Islam, M. I. (2021). Bangla Handwritten Character Recognition Using Extended Convolutional Neural Network. *Journal of Computer and Communications*, *9*(3), 158-171.

[11] Hasan, M. J., Wahid, M. F., & Alom, M. S. (2019, December). Bangla compound character recognition by combining deep convolutional neural network with bidirectional long short-term memory. In *2019 4th International Conference on Electrical Information and Communication Technology (EICT)* (pp. 1-4). IEEE.

[12] Rabby, A. S. A., Haque, S., Islam, M. S., Abujar, S., & Hossain, S. A. (2018, December). Ekush: A multipurpose and multitype comprehensive database for online off-line bangla handwritten characters. In *International Conference on Recent Trends in Image Processing and Pattern Recognition* (pp. 149-158). Springer, Singapore.

[13] Biswas, M., Islam, R., Shom, G. K., Shopon, M., Mohammed, N., Momen, S., & Abedin, A. (2017). Banglalekha-isolated: A multi-purpose comprehensive dataset of handwritten bangla isolated characters. Data in brief, 12, 103-107.

[14] Ioffe, S., & Normalization, C. S. B. Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv preprint arXiv:1502.03167*.

[15] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, *15*(1), 1929-1958.

[16] Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[17] Janocha, K., & Czarnecki, W. M. (2017). On loss functions for deep neural networks in classification. *arXiv preprint arXiv:1702.05659*.

PLAGIARISM REPORT

BANGLA HANDWRITTEN SINGLE, NUMERAL, VOWEL MODIFIER AND COMPOUND CHARACTERS RECOGNITION USING CNN

ORIGINALITY REPORT

30%	27%	18%	18%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	dspace.daffodilvarsity.edu.bd:8080 Internet Source	10%
2	www.ukessays.com Internet Source	4%
3	dokumen.pub Internet Source	2%
4	link.springer.com Internet Source	1%
5	Submitted to Queen Mary and Westfield College Student Paper	1%
6	Submitted to Daffodil International University Student Paper	1%
7	Submitted to The Scientific & Technological Research Council of Turkey (TUBITAK) Student Paper	1%
8	arxiv.org Internet Source	1%