Shahjalal University of Science and Technology

Department of Computer Science and Engineering

Bangla Handwritten Mathematical Expression Recognition

Md Mizbah Uddin Junaed

Reg: 2018331115

4^{th} year, 1^{st} Semester

Shahnur milon

Reg: 2018331100

4^{th} year, 1^{st} Semester

Department of Computer Science and Engineering

Submitted to

Mohammad Shahidur Rahman, PhD

Professor

Department of Computer Science and Engineering

29^{th} August, 2023

Abstract

The recognition of handwritten digits and symbols has long posed challenges due to their diverse shapes, sizes, and writing styles. As researchers increasingly recognize its significance in education and business, the pursuit of precise handwriting recognition gains importance. Despite prior attempts to tackle Bangla handwriting recognition, the absence of a dependable model persists. Furthermore, prevailing studies predominantly focus on digit recognition alone, without encompassing additional symbols or comprehensive mathematical expressions. In this study, we delve into the intricate realm of identifying handwritten mathematical digits and symbols within the constructs of the expressions they form. The intricate nature of the Bengali script, encompassing elements such as numerals(0-9), basic operators(addition, substraction, etc) as well as exponents and brackets, adds complexity to the task. Our research endeavors to extend the boundaries of accurate expression detection and assessment in the Bengali language context. This is achieved through a meticulously curated dataset comprising approximately 5000 instances of handwritten numerals and symbols. To facilitate the creation of an Optical Character Recognition (OCR) system in Bengali, commencing with Bangla handwritten digit recognition offers a prudent approach. The absence of a substantial and standardized dataset for Bangla digit recognition was a hurdle that existed in prior attempts. This study aims to bridge that gap by establishing a foundation for accurate digit recognition, thereby setting the stage for the subsequent development of a comprehensive OCR system tailored to the nuances of the Bengali script.

Keywords: Mathematical expression, Supervised learning, OCR, CNN, Resnet18.

Contents

Chapter 1

Introduction

Bengali is among the most widely spoken dialects on the globe, with around 220 million people using it for communication and writing. But English, Greek, and Roman letters are among the font styles used in many articles, and there are a few that worked with Bengali handwritten mathematical expression recognition.

1.1 Motivation

- **Why This Work?**

  Handwritten mathematical expression recognition in Bengali has long been overlooked, hindering effective learning experiences. By developing an accurate recognition model, we empower students and educators to effortlessly interact with mathematical content in their native language. Furthermore, this project offers a technological solution that promotes inclusivity by aiding individuals with disabilities who might struggle with conventional input methods.

- **Bridging the Language Gap :** It is a significant step towards filling the research void in recognizing handwritten mathematical expressions in the Bengali language. By addressing this gap, it effectively bridges language-specific challenges with the task of recognizing complex mathematical symbols.

- **Educational Support :** It not only benefits students and educators by accurately interpreting handwritten math expressions but also offers a valuable educational tool that enhances the comprehension and understanding of mathematical concepts in the Bengali language. The generated predicted expressions serve as a supportive resource for learners, aiding them in grasping mathematical principles with greater clarity.

1.2 Objectives

The works aims to achieve the following objectives:

Create a comprehensive and diverse dataset of handwritten Bengali mathematical expressions encompassing digits, operators, exponents, and brackets. Design and implement an Artificial Neural Network architecture for recognizing and predicting handwritten mathematical symbols.

Train the model on the dataset to achieve a high level of accuracy and robustness in symbol recognition and expression prediction. To design, implement, and fine-tune an ANN-based system.

A crucial objective is to curate a meticulously designed dataset comprising around 5000 instances of handwritten numerals and symbols in the Bengali script. This dataset will serve as a foundation for training and evaluating machine learning models for accurate character recognition.

Chapter 2

Related Works

While previous works have laid the foundation for mathematical expression recognition and OCR, there is a noticeable paucity in research addressing Bengali mathematical expressions. The extant studies predominantly revolve around English language expressions, utilizing methods such as CNNs and RNNs for image analysis. Bridging the gap between these advancements and the specific challenges of Bengali script constitutes a novel contribution. The ability to recognise handwritten mathematical expression has come a long way. Three distinct sub-sections make up the literature review part. Works based on optical character recognition are covered in the first subsection, whereas papers on digit recognition are detailed in the second. The works based on handwritten mathematical expression recognition are contained in the third subsection.

2.0.1 OCR Based Papers

The study [1] by Jamshed Memon et al. thoroughly reviewed 176 papers on handwritten OCR that were published between 2000 and 2019 in order to conduct a structured literature study. According to this study, the OCR gives priority to six different languages. After executing a thorough review approach using electronic resources, it has been found that, while published research papers have proposed a number of alternative OCR systems, a significant area still required progress like data analysis commercialization.

This research [2] by Noman Islam et al. incorporates data from several previously published OCR studies. The pre-processing, symbol segmentation, feature extraction, classification, and other processes required for a good OCR system were all outlined.

2.0.2 Digit Recognition Based Papers

The work [3] used autoencoder and deep CNN to recognise Bangla handwritten digit. Existing research accentuates the diversity of numeral styles across languages, affecting model accuracy. This study breaks ground by employing unsupervised pre-training alongside deep ConvNets for recognizing Bangla digits. CMATERDB 3.1.1 and ISI datasets serve as evaluation benchmarks. The research's novelty lies in achieving a notable 99.50% accuracy, marking a significant advancement in handwritten Bangla digit recognition.

The study [4] by Muntarin Islam et al. proposed a handwritten character system consisting of ResNet50, Inception-v3 and EfficientNet-b0 with NumtaDB dataset consisting of 17,000 examples of handwritten Bengali digits having 10 classes. The result of their models is quite impressive, achieving 97% accuracy mostly in 10 categories. This study also contrasts various pre-trained algorithms for Bengali handwritten digit recognition systems. Testing accuracy they achieved for InceptionV3, EfficientNet-b0 and RestNet50 are 88%, 96%, and 94%.

In this study [5], the recognition of Bangla digits is done using NumtaDB, a sizable and objective dataset. The NumtaDB collection presents difficulties due to the abundance of enhanced and raw photos. In this article, deep convolutional neural network (CNN) is utilised as the classification model. Images are processed using several preprocessing approaches.

2.0.3 Handwritten Mathematical Expression Recognition Based Papers

This article [6] by Md Bipul Hossain et al. shows a process of identifying and dealing with a single mathematical problem as well as a group of scrawled math questions. They created several samples and also utilized a customized MNIST dataset. One grayscale image of 32*32 pixels was utilized for the dataset. Initial preparation of such dataset photos included grayscale transformation, noise removal, and binarization with just an adjustable threshold.

The primary objective of the paper [7] was to identify HME utilizing CNN. The images were preprocessed utilizing grayscale conversion, noise reduction that uses the average, binarization using cutoff, and flattening to maintain the depth of the foreground after the HME database was assembled by a number of researchers. Single pixel to ensure that the picture may be processed clearly. They used CNNâs 83 different classifications as the classifier. The reliability of just this model was roughly 87.72 %.

Chapter 3

Methodology

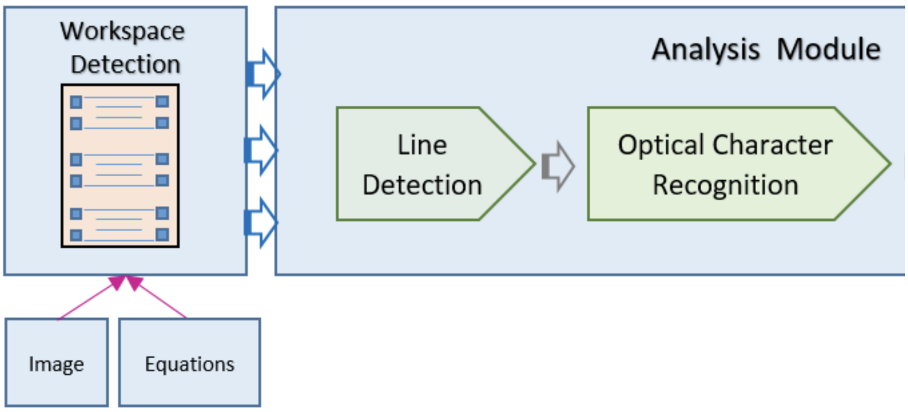

Figure 3.1: Overview of the workflow

3.1 Dataset preprocessing

3.1.1 Data Collection

We collected a dataset of around 5000 digits and symbols.

Eliminate Noise: Remove extraneous characters, symbols, and emojis that might not be helpful for the translation process.

Quality Control: Manually review a portion of the data to find and eliminate any sentences that have poor translations or grammatical problems.

3.1.2 Data Splitting and Augmentation

Shuffling: Shuffle the dataset to ensure randomness during training.

Training and Validation Split: The dataset is split into training and validation subsets in the ratio of 8:2 using tensorflowâs image data generator.

Dataset Statistics

Calculate basic statistics such as the number of samples, average sentence length, and vocabulary size to gain insights into your dataset's characteristics.

Two open source datasets such as MNIST and Kaggleâs mathematical symbols are used for optical character recognition.

Total no of images = 6,250

Collected more handwritten digits and symbols = 5,000

No of classes = 15

Classes = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9', '+', '-', 'times','(',')']

3.2 System Architecture

As shown the overall solution can be divided into two parts, i.e âWorkspace Detectionâ module and âAnalysis Moduleâ.

3.2.1 Modules and Major Approaches

Workspace detection module is responsible for detecting multiple workspaces in a given sheet of paper using pre-defined markers sheet.

Analysis module is responsible for detecting and localizing characters in any given single workspace, and mathematically analysing them and drawing red, green lines depending upon their correctness

Workspace Detection Module

Workspace detection model assumes that there are valid rectangular boxes in the given scanned worksheet. Steps involved in workspace detection are

1. Finding closed object contours (Rectangular boxes)
2. Sorting the contours (Top-to-Bottom) based on the coordinates
3. Choosing the desired boxes based on the area

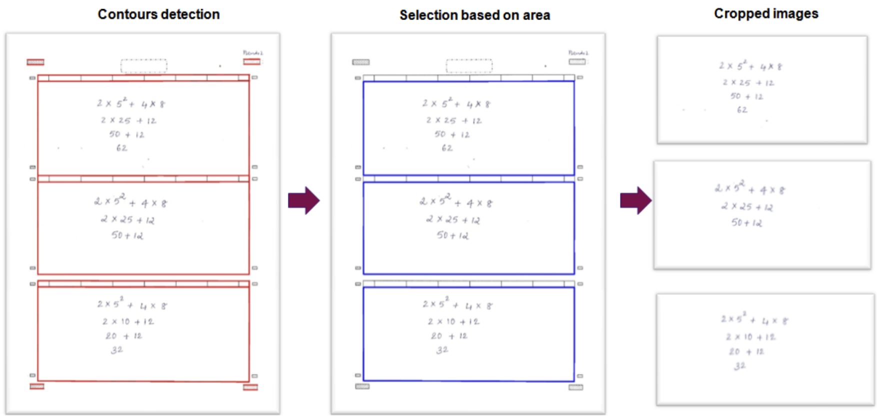

Figure 3.2: Workspace detection steps

Analysis Module Approaches

Line Detection using forward derivative: The approach for line detection assumes that there is a sufficient gap between lines and there is some intersection between exponential characters and line. The binary images of the detected workspaces are compressed in a single array to take forward derivative thereby detecting the coordinates of each line.

Line Detection using OCRopus: OCRopus is a collection of document analysis programs, it performs the following steps

1. Binarization
2. Page-Layout Analysis
3. Text-Line Recognition
4. OCR

The default parameters and settings of OCRopus assume 300dpi binary black-on-white images. If images are scanned at a different resolution, the simplest thing to do is to downscale/upscale them to 300dpi. The text line recognizer is fairly robust to different resolutions, but the layout analysis is quite resolution dependent.

Character Recognition Approaches

Deep Columnar Convolutional Neural Network : The proposed model is a single deep and wide neural network architecture that offers near state–of-the-art performance like ensemble models on various image classification challenges, such as MNIST, CIFAR-10 and CIFAR-100 datasets.

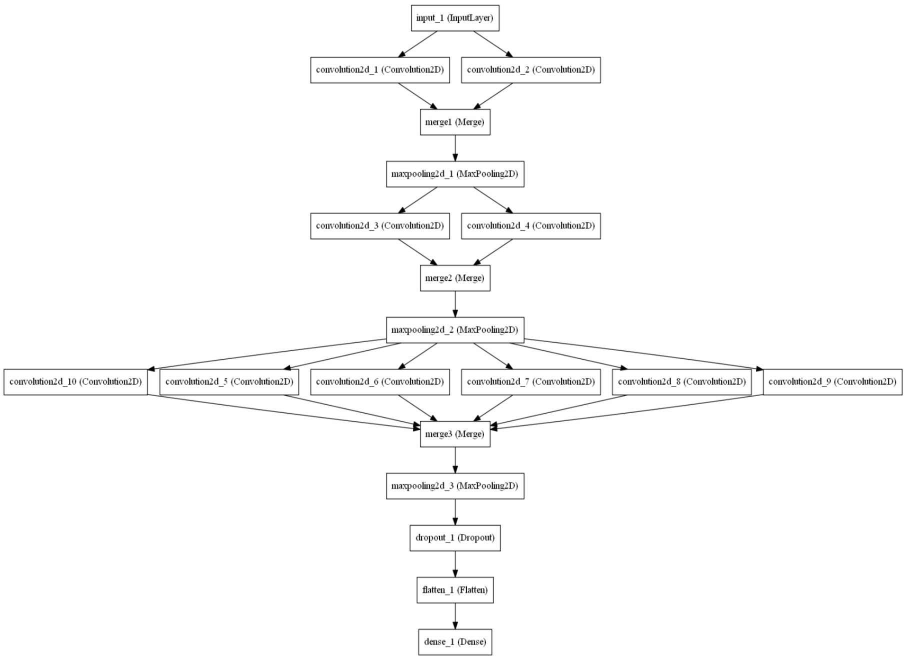

Figure 3.3: DCCNN Architecture

Training:

- Activation Function Softmax is used in final layer, others contain ReLu activation function.
- Optimizers: AdaDelta optimizer is an adaptive learning rate method which requires no manual tuning of a learning rate performed well compared to other optimizers. The parameters:
- **Learning Rate = 1.0** and **Rho = 0.95 (decay factor)**
- Batch Normalization is used to overcome vanishing gradient problem
  Dropout of 50 % is used before the final dense layer
- Model is trained for only 10 epochs with accuracy up to 96 % Augmentation only slightly improved accuracy in this case (Random rotations, width shift, height shift)

Chapter 4

Result and Discussion

The model's performance is evaluated using metrics such as accuracy, precision, recall, and F1-score. Quantitative analysis reveals the effectiveness of the proposed approach in recognizing individual symbols and predicting complete expressions.

Hyperparameter Settings: The learning rate and batch size for training CNN and ResNet18 were identical for both models. The batch size was 64, and the learning rate was 0.001. We applied 10 epochs for CNN, while we used 73 epochs for ResNet18 and obtained an accuracy more than 95% for both.

Performance Metrics: The most logical performance metric is accuracy, which is just the proportion of properly predicted observations to all observations. In terms of positive observations, precision is the proportion of accurately anticipated observations to all predicted positive observations. A good classifierâs precision should preferably be 1 (high). Recall (sensitivity) is the proportion of accurately predicted positive observations to all of the actual class observations. The ideal recall for a good classifier is 1 (high). The weighted average of Precision and Recall is the F1 Score. Therefore, both false positives and false negatives are considered while calculating this score. Although F1 is generally more beneficial than accuracy, especially if you have an uneven class distribution, it is not intuitively as simple to understand as accuracy.

Discussion:

The project's results highlight the model's proficiency in symbol recognition and expression prediction, even in the context of complex mathematical expressions. Challenges related to varying handwriting styles and noise in the dataset were addressed through data preprocessing techniques.

Chapter 5

Conclusion

Due to the language's extensive use of complicated characters and signs, the identification of Bengali handwritten characters and signs is a challenging research area. By using the same detection and segmentation steps as in the suggested technique, complex problems like evaluating and solving equations may be solved. Our work examined how the cutting-edge CNN and Restnet-18 architectures recognise handwritten Bengali phrases. The ResNet18 strategy has performed much better than the CNN architecture. Studies using four sizable datasets for Bengali handwritten digit recognition demonstrate the efficacy of these 2 models. In the subsequent study, this segmentation approach may be applied more skillfully to significantly improve the results. We anticipate that this research will have a significant impact on the OCR market. By using the same detection and segmentation steps as in the suggested technique, complex problems like trigonometric equations may be solved. But it won't be correctly detected if there isn't enough space between the numerals and the lines. Furthermore, if the exponential digit is somehow not clearly expressed, it will be incorrectly detected. We anticipate that this research will have a significant impact on the OCR market. In conclusion, the developed ANN-based model showcases promising potential for Bengali Handwritten Mathematical Expression Recognition. The work not only contributes to the underrepresented domain of Bengali language-specific ML applications but also opens avenues for further research in diverse languages and complex expression recognition. The successful creation of the model and its ability to accurately predict expressions demonstrate its utility in educational, accessibility, and real-world scenarios.

References

[1] J. Memon, M. Sami, R. A. Khan, and M. Uddin, “Handwritten optical character recognition (ocr): A comprehensive systematic literature review (slr),” *IEEE Access*, vol. 8, pp. 142 642–142 668, 2020.

[2] N. Islam, Z. Islam, and N. Noor, “A survey on optical character recognition system,” *arXiv preprint arXiv:1710.05703*, 2017.

[3] M. Shopon, N. Mohammed, and M. A. Abedin, “Bangla handwritten digit recognition using autoencoder and deep convolutional neural network,” in *2016 International Workshop on Computational Intelligence (IWCI)*. IEEE, 2016, pp. 64–68.

[4] S. Das, M. S. Imtiaz, N. H. Neom, N. Siddique, and H. Wang, “A hybrid approach for bangla sign language recognition using deep transfer learning model with random forest classifier,” *Expert Systems with Applications*, vol. 213, p. 118914, 2023.

[5] A. Shawon, M. J.-U. Rahman, F. Mahmud, and M. A. Zaman, “Bangla handwritten digit recognition using deep cnn for large and unbiased dataset,” in *2018 international conference on Bangla speech and language processing (ICBSLP)*. IEEE, 2018, pp. 1–6.

[6] M. B. Hossain, F. Naznin, Y. Joarder, M. Z. Islam, and M. J. Uddin, “Recognition and solution for handwritten equation using convolutional neural network,” in *2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR)*. IEEE, 2018, pp. 250–255.

[7] L. Dâsouza and M. Mascarenhas, “Offline handwritten mathematical expression recognition using convolutional neural network,” in *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*. IEEE, 2018, pp. 1–3.