An overview: Optical Character Recognition Systems on Bangla Scripts

MD Tanjim Mostafa*,
Ehsanur Rahman Rhythm, Md Humaion Kabir Mehedi, and Annajiat Alim Rasel
Department of Computer Science and Engineering (CSE)
School of Data and Sciences (SDS)
Brac University
66 Mohakhali, Dhaka - 1212, Bangladesh

***Abstract*—OCR systems are very powerful tools that are used to convert handwritten texts or digital data on an image to machine readable texts. The importance of Optical Character Recognition (OCR) for handwritten documents cannot be overstated due to its widespread use in human transactions. OCR technology allows for the conversion of various types of documents or images into machine understandable data that can be analyzed, edited, and searched. In earlier years, manually crafted feature extraction techniques were used on comparatively small datasets which were not good enough for practical use. With the advent of deep learning, it was possible to perform OCR tasks more efficiently and accurately than ever before. In this paper, several OCR techniques have been reviewed. We mostly reviewed works on Bangla scripts and also gave an overview of the contemporary works and recent progresses in OCR technology(e.g. trOCR,transformer w/ CNN). It was found that for Bangla handwritten texts, CNN models like DenseNet121, ResNet50, MobileNet etc are the commonly adopted techniques because of their state of the art performance in object recognition tasks. Using an RNN layer like LSTM or GRU alongside the base CNN based architecture, the accuracy can be further improved. TrOCR is a fairly new technique in this field that shows promise. Experimental results show that, On synthetic IAM handwriting dataset it showed a CER of 2.89. The goal of this paper is to provide a summary of the research conducted on character recognition of handwritten documents in Bangla Scripts and suggest future research directions.**

***Index Terms*—Optical Character recognition, BanglaOCR, Deep learning, Bangla handwriting.**

I. Introduction

Optical character recognition (OCR) is the technique of automatically extracting text from images or scanned documents and converting it into machine-readable formats. OCR can be performed on digitally printed images or images with handwritten words. OCR systems are particularly interesting because they allow us to digitize traditional documents. Digital data has various advantages. For once, they are easy to manipulate which helps with the entry of new data and the modification of the existing document. Second, digital data help process the document, allowing us to perform tasks like search, context extraction, and translation of the document. OCR systems can be used for automation purposes in libraries, post offices, educational institutes, etc. Because of its' usefulness, a lot of research has been done in several domains of OCR. Despite that, optimally and accurately recognizing Bangla handwritten characters still remain a daunting task because of how diverse and complicated the Bangla language is. One of the earliest work in Bangla OCR was done by Ray et al., [1] where they used a simple nearest-neighbor classifier for recognizing hand-printed Bengali alphabetic characters. Since then the field of OCR research has progressed a lot. Most of the recent works have been done using Deep neural network models as the base architecture. Our paper explores different techniques of performing OCR on Bangla handwritten text to find out the optimal approach. [tentative]

II. Literature review

In this section we review research papers that have been done in the domain of OCR, mostly for Bangla scripts and also contemporary works in other languages. Liu and Suen [2] introduced directional gradient features for handwritten Bangla digit classification using ISI Bangla numeral dataset which has 19,392 training samples and 4000 test samples for 10 Bangla numerical characters [3]. Hasnat et al. [4] presented a domain specific Handwritten Character Recognition (HCR) system with the capability of classifying both printed and handwritten characters. Their approach involved the application of Discrete Cosine Transform (DCT) to the input image, followed by character classification using Hidden Markov Model (HMM) techniques. Das et al. [5] introduced a feature set representation to recognize handwritten Bangla Characters. They used a combination of modified shadow features, octant and centroid features, distance based features, quad tree based longest run features achieving an accuracy of 85.40% on a dataset consisting of 50 character classes. However, these methods relied on manually supervised features created from small datasets, making them impractical for real world applications. With the advent of deep neural networks, that problem was partially mitigated. Safir et al. [6] proposed an end to end OCR architecture that depends on CTC loss function. They used convolutional neural network architectures as feature extractors and then they passed the extracted features through a recurrent layer(LSTM, GRU). Finally, a fully connected layer was used to generate the probability distribution of the final prediction. Chatterjee, Dutta et al. [7] proposed a transfer learning technique on Deep convolutional neural network, namely ResNet50. Rabbani Alif et al. [8] proposed

a modified ResNet-18 architecture by adding a dropout layer to each module which in turn improved the generalization and regularization of the input data. They applied their model on two isolated Bangla handwritten dataset namely BanglaLekha-isolated [9] and CMATERdb dataset [10] achieving 95.10% and 95.99% accuracy respectively. Alom et al. [11] used different CNN models on three different datasets from CMATERdb for Bangla handwritten digits, alphabets and special characters and reported that DenseNet showed the best performance. Dipu, Shohan et al. [12] used three deep neural network based image classification models namely Inception V3, VGG16, and Vision Transformer. Their main goal was to mitigate the problem of previous systems struggling with recognizing Bengali compound words. They have shown a 98.65% accuracy with their best model as VGG-16.

CNNs have proven to be highly successful in performing OCR tasks in other languages as well. [13] have showed that multi-column Deep Neural Networks achieved recognition rates of Chinese characters comparable to human accuracy. The implemented a 11-layer deep neural network with hundreds of maps per layer and trained the model on raw, distorted images to prevent overfitting. Kim et al. [14] proposed a CNN based model that achieved state of the art accuracy for recognizing hangul characters. However, transformer based models have become a widely adopted technique for performing Natural Language Processing(NLP) tasks.

Although, Transformer based systems are fairly new in the field of OCR and we haven't found any Bangla OCR systems based on transformers. Experimental results have shown that TrOCR achieves extraordinary results on printed, handwritten and scene text recognition with just a simple encoder-decoder model, without needing any post-processing steps [15].

III. Basics of Bangla Scripts

Bangla language consists of 50 letters in total. Among them, 11 are vowels and 39 are consonants. Two or more characters combine to form a new compound character called Juktoborno. Bangla characters are very diverse in shape which makes them difficult to deal with in OCR systems. There are 10 modifiers and 10 numeric characters as well. Bangla language has a number of distinctive features, including the use of loops and hooks on some letters, also the use of diacritical marks above letters called 'matra' and diacritical marks below letters called 'hoshonto'. Some of these diacritical marks indicate distinct character pronunciation.

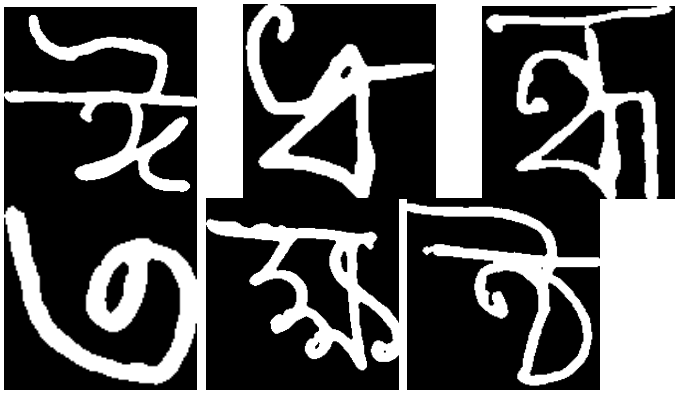

Fig. 1. Some Bangla characters

IV. Data Collection And Pre-processing Techniques

A. Datasets

The most commonly used dataset for Bangla OCR models is the one called BanglaLekha-Isolated. It consists of handwriting samples of 50 Bangla basic characters, 10 Bangla numerals and 24 selected compound characters. There are 2000 handwriting samples for each of the 84 characters. The samples were collected from subjects of different age groups and gender [9]. In 2021, Mridha et al. created a new bangla handwritten dataset called BanglaWriting which contains single page handwriting of 260 individuals of different age groups [16]. Although a lot smaller in size, it's similar to the popular IAM handwriting dataset [17]. that is often used to train and test transformer based OCR models. The dataset has manually-generated word labels and bounding boxes for each word along with the Unicode representation of the writing. The dataset is suitable for machine learning models, deep learning models, producing embedding vectors of handwriting, etc. CMATERdb is a collection of handwritten character datasets primarily used for research and development in the field of pattern recognition and machine learning. They have separate datasets for Bangla numerical digits,Alphabets, special characters and also compound characters. Each characters in these CMATERdb datasets are labeled with a corresponding class label. The CMATERdb 3.1.1 dataset [18] for digits contains 6000 images of unconstrained isolated handwritten Bnagla numerals. Whereas, CMATERdb dataset for compound characters [10] consists of 171 character classes. Altogether, 55,278 isolated character images, belonging to 199 different pattern shapes, were collected using three different data collection modalities. The database is divided into training and test sets in 4:1 ratio for each pattern class.

B. Data Pre-Processing

Several pre-processing techniques can be performed to make the data more suitable for a specific task and also to bring more variation to the original data. Safir et al. reshaped and normalized the word-level images from BanglaWriting dataset

to fulfill some pre-defined conditions and then performed augmentation techniques like Horizontal cutout, Vertical cutout, Gaussian noise etc. to create a more diverse dataset [6].

V. Methodology

A. Understanding Neural network based OCR systems

In the field of machine learning and computer vision, implementing Deep neural network(DNN) has been a popular technique for a while. Due to having multiple layers, DNN methods are very capable of representing the highly varying nonlinear function compared to shallow learning approaches. DNNs can easily learn hierarchical representations of visual features from raw input data which makes them more suitable for Optical Character Recognition. The lower and middle layers of a DNN are used for feature extraction of an image and the higher layers are used for performing classification tasks using those extracted features. This integration of layers within a single network allows us to build an end-to-end framework. One of the DNN architectures, Convolutional neural network(CNN) is most used in OCR systems as the underlying architecture. Most of the recent OCR systems built or optimized for Bangla language that we have reviewed are based on CNN architecture as well. (Alom et al., 2018) have suggested that DCNN models show superior performance compared to other popular object recognition approaches due to their ability to extract discriminative features from raw data and represent them with a high degree of invariance [11]. The overall architecture of a CNN consists of two main parts: feature extractor and classifier. In feature extractor,each layer of the network receives the output from its immediate previous layer as inputs and passes the current output as inputs to the immediate next layer, whereas the classification part generates the predicted outputs associated with the input data [11]. The architecture can have multiple layers including convolution layers, pooling layers and fully connected layers. There are several CNN models that are commonly used in the field of computer vision and object classification like VGG-16,AlexNet, VGG Net, GoogleNet, ResNet, DenseNet etc. each of these models serves different purposes in terms of their design and implementation. For instance, ResNet and GoogleNet are specifically tailored for large-scale applications, while VGG Net has a more general architecture. FractalNet offers an alternative to ResNet, and DenseNet stands out due to its unique unit connectivity, where each layer is directly connected to all subsequent layers.

B. Understanding transformer based OCR systems

Generally, OCR systems consist of two main modules.

1) Text detection module
2) Text recognition module

Text detection module tries to detect the text blocks that exist in the source image either in word level or the text line level. This task can be thought of as an object detection problem, but instead of detecting objects in images, we are detecting text blocks in documents. The Text Recognition module is responsible for understanding the content of the detected text block and converting the visual signals into natural language tokens. The key difference between conventional Convolutional neural network(CNN) based OCR systems and transformer based OCR system is that the former models are built on top of existing CNN models for image understanding and recurrent neural network(RNN) for character level text generation whereas in the latter system a text line is passed to an image based encoder and then the output from the encoder is passed to a text based BERT-style decoder model.

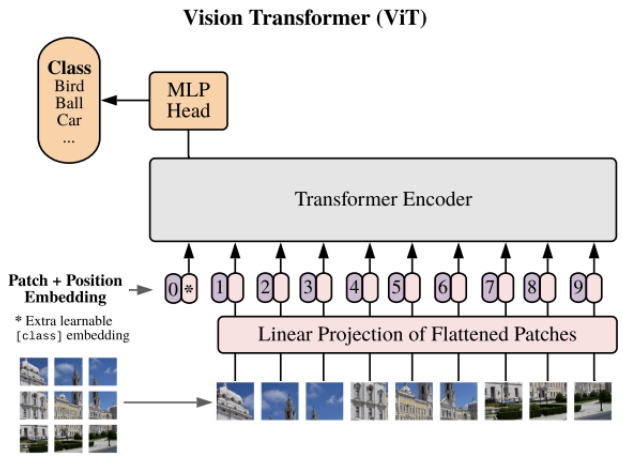

Fig. 2. Illustration of a transformer Encoder

1) Encoder: The source image is divided into small, equally-sized patches, treating them as individual elements similar to words in a sentence. Each patch is flattened into a one-dimensional vector and combined with positional embeddings. This embeddings are then passed through transformer encoder layers to process the image data. In OCR, the image is a series of localized text boxes. In order to maintain consistency across localized text boxes, the images or the text blocks on the images are reshaped to a uniform height and width (H x W). The image is then decomposed into patches of a specified size where the patch size is HW/(PxP). Here, P is the patch size. Each patch is then flattened and linearly projected to a D-Dimensional vector to create patch embeddings. These embeddings, together with two specialized tokens are then given learnable 1D position embeddings according to their absolute positions. Then the input sequence is passed through a series of encoder layers, each consisting of a multi-head self-attention module and a fully connected feed-forward network. After that, residual connections and layer normalization follows to create a more robust and stable representation of the input data [19].

2) Decoder: The derived output from the ViTransformers are passed through the Decoder which is basically a text transformer like BERT, RoBERTa etc. The decoder module has stacks of identical layers to the encoder transformer. In the decoder, a module called ”encoder-decoder attention” is used between the multi-head self-attention and feedforward network

[15]. This module is responsible for distributing attention differently across the output of the encoder. In this module, the keys and values are derived from the encoder output, while the queries are taken from the decoder input. Then the embeddings from the decoder are projected from the model dimension to the dimension of vocabulary size. The softmax function calculates the probabilities over the vocabulary.

VI. Discussion And result Analysis

Safir, Ohi et al. used end-to-end system where they used popular neural network architectures like DenseNet121, Xception, NASNet etc to extract features from the training data and then they passed the output through an RNN layer namely LSTM or GRU layer [6]. They applied their model on the BanglaWriting [16] dataset counted character error rate and word error rate to evaluate the performance of each model and found out that DenseNet121 with GRU layer achieves the best performance with CER of 0.09 and WER of 0.273. Alom et. al [11] have showed that DenseNet had the best performance on all three of the datasets used(digit, alphabet, special characters) with accuracy 99.13%,98.31%,98.18% respectively. Other Deep CNN methods like VGG Net,FractalNet, ALL-Conv etc, had inferior performance in comparison. Comparing different models used in [20], [21], [8], [7] we found that vanilla CNN had an accuracy of 89.01% using 50 classes, Ensemble CNN had an accuracy of 97.21% using 84 classes, ResNet-18 had an accuracy of 95.10% using 84 classes and ResNet-50 achieved an accuracy of 96.12% on the validation set using 84 classes without using Ensemble Learning. All of these models were applied on BanglaLekha-isolated dataset [9].

Method	Dataset	Accuracy (%)
DenseNet	CMATERdb 3.1.1 Digit	99.13
DenseNet	CMATERdb 3.1.1 Alphabets	98.31
DenseNet	CMATERdb 3.1.1 Special Characters	98.18
Vanilla CNN	BanglaLekha-Isolated 50 Classes	89.01
ResNet-18	BanglaLekha-Isolated 84 Classes	95.10
Ensemble CNN	BanglaLekha-Isolated 84 Classes	97.21
ResNet-50	BanglaLekha-Isolated 84 Classes	96.12
VGG-16	BanglaLekha-Isolated 84 Classes	98.65
VisionTransformer	BanglaLekha-Isolated 84 Classes	96.88

TABLE I
Performance comparison of several DNN models

TrOCR on the other hand doesn't require conventional CNN models for image understanding. Instead, it uses an image transformer model as the visual encoder and a text transformer model as the text decoder. Also, the wordpiece is used as the basic unit for the recognized output instead of the character-based methods which reduces the computational cost associated with additional language modeling [15]. Li et al. trained their trOCR model on the popular IAM handwriting dataset [17] and achieved an astonishing CER of 2.89 without even leveraging any human-labeled data.

From the results presented, we can conclude that CNN architectures are the most accurate tried and tested method of performing Bangla OCR tasks. However, there is still a lack of quality datasets that contains sufficient samples of all the characters in Bangla Language. [7] have complained that they found quite a few datapoints that were mislabeled and argued that their model would have had better accuracy if those mislabeled datapoints were removed.

Authors	Dataset	Model	Error Rate
safir et al.	BanglaWriting	Xception+GRU	CER= 5.65
		Xception+LSTM	CER= 4.61
		DenseNet+GRU	CER=0.91
		DenseNet+LSTM	CER=1.44
Ahmed et al.	50 documents of	Edit Distance	CER= 1.68
	Bangla Scripts	N-gram	CER= 1.72
Diaz et al.	Internal+IAM	S-Attn/ CTC	CER= 2.75
		Transformer w/ CNN	CER=2.96
Li et al.	Synthetic	TrOCR Large	CER=2.89
	IAM handwriting	TrOCR Small	CER=4.22

TABLE II
RESULT ANALYSIS OF DIFFERENT METHODS

VII. Conclusion

In this paper, we have reviewed several papers on OCR systems, predominantly on Bangla Scripts. We found that CNN based architectures are commonly adopted because of their high performance with object recognition related tasks. However, transformer based models show promise in solving OCR related problems for efficiently and accurately. Future research efforts in BanglaOCR should focus more on finding ways to implement transformer based systems for Bangla scripts. There's still a need for good labeled datasets of Bangla characters, words and sentences. Our hope is that this survey encourages researchers to work on mitigating the limitations in Bangla OCR research and make new advancements in fields that haven't been explored yet.

References

[1] A. Ray and B. Chatterjee, "Design of a nearest neighbour classifier system for bengali character recognition," *IETE Journal of Research*, vol. 30, no. 6, pp. 226–229, 1984. [Online]. Available: https://doi.org/10.1080/03772063.1984.11453273

[2] C.-L. Liu and C. Suen, "A new benchmark on the recognition of handwritten bangla and farsi numeral characters," *Pattern Recognition*, vol. 42, pp. 3287–3295, 12 2009.

[3] B. B. Chaudhuri, "A Complete Handwritten Numeral Database of Bangla – A Major Indic Script," *Suvisoft*, Oct. 2006. [Online]. Available: https://hal.science/inria-00104486

[4] Md. A. Hasnat, S. M. M. Habib, and M. Khan, "A High Performance Domain Specific Ocr For Bangla Script," in *Novel Algorithms and Techniques In Telecommunications, Automation and Industrial Electronics*. Dordrecht, The Netherlands: Springer, 2008, pp. 174–178.

[5] N. Das, S. Basu, R. Sarkar, M. Kundu, M. Nasipuri, and D. kumar Basu, "An improved feature descriptor for recognition of handwritten bangla alphabet," 2015.

[6] F. B. Safir, A. Q. Ohi, M. Mridha, M. M. Monowar, and M. A. Hamid, "End-to-end optical character recognition for bengali handwritten words," in *2021 National Computing Colleges Conference (NCCC)*, 2021, pp. 1–7.

[7] S. Chatterjee, R. K. Dutta, D. Ganguly, K. Chatterjee, and S. Roy, "Bengali Handwritten Character Classification Using Transfer Learning on Deep Convolutional Network," in *Intelligent Human Computer Interaction*. Cham, Switzerland: Springer, Apr. 2020, pp. 138–148.

[8] M. Al Rabbani Alif, S. Ahmed, and M. A. Hasan, "Isolated Bangla handwritten character recognition with convolutional neural network," in *2017 20th International Conference of Computer and Information Technology (ICCIT)*. IEEE, Dec. 2017, pp. 1–6.

[9] M. Biswas, R. Islam, G. K. Shom, M. Shopon, N. Mohammed, S. Momen, and M. A. Abedin, "Banglalekha-isolated: A comprehensive bangla handwritten character dataset," 2017.

[10] N. Das, K. Acharya, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, "A benchmark image database of isolated Bangla handwritten compound characters," *IJDAR*, vol. 17, no. 4, pp. 413–431, Dec. 2014.

[11] M. Z. Alom, P. Sidike, M. Hasan, T. M. Taha, and V. K. Asari, "Handwritten bangla character recognition using the state-of-the-art deep convolutional neural networks," *Computational Intelligence and Neuroscience*, vol. 2018, p. 1–13, 2018.

[12] N. M. Dipu, S. A. Shohan, and K. M. A. Salam, "Bangla optical character recognition (ocr) using deep learning based image classification algorithms," in *2021 24th International Conference on Computer and Information Technology (ICCIT)*, 2021, pp. 1–5.

[13] D. Cireşan and U. Meier, "Multi-column deep neural networks for offline handwritten chinese character classification," in *2015 International Joint Conference on Neural Networks (IJCNN)*, 2015, pp. 1–6.

[14] I.-J. Kim and X. Xie, "Handwritten Hangul recognition using deep convolutional neural networks," *IJDAR*, vol. 18, no. 1, pp. 1–13, Mar. 2015.

[15] M. Li, T. Lv, J. Chen, L. Cui, Y. Lu, D. Florencio, C. Zhang, Z. Li, and F. Wei, "TrOCR: Transformer-based Optical Character Recognition with Pre-trained Models," *arXiv*, Sep. 2021.

[16] M. Mridha, A. Q. Ohi, M. A. Ali, M. I. Emon, and M. M. Kabir, "BanglaWriting: A multi-purpose offline bangla handwriting dataset," *Data in Brief*, vol. 34, p. 106633, feb 2021.

[17] U.-V. Marti and H. Bunke, "The iam-database: An english sentence database for offline handwriting recognition," *International Journal on Document Analysis and Recognition*, vol. 5, pp. 39–46, 11 2002.

[18] N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, "A genetic algorithm based region sampling for selection of local features in handwritten digit recognition application," *Appl. Soft Comput.*, vol. 12, no. 5, pp. 1592–1606, May 2012.

[19] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," *arXiv*, Oct. 2020.

[20] B. Purkaystha, T. Datta, and M. S. Islam, "Bengali handwritten character recognition using deep convolutional neural network," in *2017 20th International Conference of Computer and Information Technology (ICCIT)*, 2017, pp. 1–5.

[21] S. Saha and N. Saha, "A Lightning fast approach to classify Bangla Handwritten Characters and Numerals using newly structured Deep Neural Network," *Procedia Comput. Sci.*, vol. 132, pp. 1760–1770, Jan. 2018.