DOI: https://doi.org/10.36094/sc.v89.2023.Deep_Convolutional_Transfer_Learning.De_Verma.182

ARTICLE

Sci. and Cult. 89 (5–6) : 182-187 (2023)

DEEP CONVOLUTIONAL TRANSFER LEARNING APPROACH FOR BENGALI HANDWRITTEN CHARACTER RECOGNITION FROM DOCUMENT IMAGE

SHANKHA DE[1] AND DIPTI VERMA[2]

Handwritten character recognition from a document image is no-doubt a challenging task. Optical Character Recognition (OCR) research success has limited to very few scripts like Roman, English and Chinese. Bengali literature is very rich and voluminous. For digitization of handwritten manuscripts, OCR needs higher recognition rate. In this paper, a transfer learning approach is applied. A customized convolution neural network with 34 layers model is applied upon Bengali handwritten font image CMATERDb3.1.2 dataset and gives recognition results of 98.96% accuracy.

Introduction

Bengali is one of the most popular official languages in India. Bengali literature is very rich and voluminous mostly contributed by the work of poets and writers like Rabindranath Tagore, Michael Madhusudan Dutt, Bankim Chandra Chattopadhyay, Sarat Chandra Chattopadhyay, Kazi Nazrul Islam. Regeneration version of their poems, shyama sangeet, sonnets, novels, short stories justifies the need to formulate still unexplored task of Bengali language alphabet recognition system. Bengali Script recognition is a challenging area of research since a couple of decades. Handwritten character recognition is required to convert historical documents, letters, diaries and many other manuscripts in digital form. Proper selection of features and classifiers are also important task for recognition process. Some of the research works have demanded developing Bengali optical character recognition system with very good success rate. Here some of these approaches are listed and tried to find out the problems associated with developing OCR system mainly for handwritten Bengali fonts. This paper also proposed a supervised transfer learning method to improve the accuracy in this domain. This paper is organized as follows: Section 2 gives brief description and application of Bengali Script. Section 3 reviewed recent research work in Bengali HWR process using convolution neural network. Section 4 describes the proposed model. Section 5 shows the results followed by the conclusion of the research work.

Overview of Bengali Script

Bengali script is derived from ancient Brahmin script. Bengali characters can be divided into vowels, consonants, compound characters, modifiers and punctuations. It has 11 vowels and 39 consonants. 10 Vowels when attached with consonant sometime get special shape called modifiers and more than one consonant combined together makes compound characters. There are more than 250 compound characters composed of 2, 3 or 4 consonants[1]. There are 12 vowel out of which one vowel (lli ঌ) becomes obsolete and 39 consonant characters in modern *Bangla* alphabet as shown in Table 1 and Table 2. They are called *basic characters*. Out of these fifty characters , 'Khanda-ta - ৎ' , 'Anusvara - ং ' , 'Visarga - ঃ' and 'Chandra-bindu - ঁ' is used as a modifier for contextual purpose. It is normally written from left to write mostly following a header line called 'matra' which is normally used for segmentation of

1 Department of Computer Science and Engineering, Bhilai Institute of Technology, Durg
e-mail: shankhada2009@gmail.com,
2 Chhattisgarh Swami Vivekanand Technical University Bhilai, e-mail: diptiverma.sac@gmail.com

Table 1. Bengali Vowels

1	2	3	4	5	6	7	8	9	10	11	12
অ	আ	ই	ঈ	উ	ঊ	ঋ	ঌ	এ	ঐ	ও	ঔ

Table 2. Bengali Consonants

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
ক	খ	গ	ঘ	ঙ	চ	ছ	জ	ঝ	ঞ	ট	ঠ	ড	ঢ	ণ	ত	থ	দ	ধ	ন
21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	
প	ফ	ব	ভ	ম	য	র	ল	শ	ষ	স	হ	ড়	ঢ়	য়	ৎ	ং	ঃ	ঁ	

words. There are no upper and lower case in Bangla language[2].

There are various application domains where handwritten character recognition has major role. Some of these areas are postal address interpretation to identification of city, pin code and other information from postal address, Bank cheque processing, signature authentication, author identification; identify authors from the handwritten manuscript and medical transcription**.**

Literature Review

CNN model is one of the most successful model in handwritten character recognition in comparison with SVM, HMM or syntactic pattern matching and other variation of gen3eral neural network . It is applied in many other languages like English, Russian, Chinese, Arabic and other European languages[3-11]. Sometimes classification features are extracted and then CNN is applied. In this section various recent research papers are reviewed where CNN is applied in Bengali language for character recognition. Table 3 shows various existing approaches towards recognition of Bengali handwritten character dataset.

So it is observed that CNN models are most successful in recognizing Bengali handwritten characters. Various CNN models are applied. Some researchers achieved more accuracy rate using CNN by their own dataset. This paper tries to implement another variation of CNN model, for Bengali handwritten character recognition to achieve higher accuracy.

Methodology

In this paper a supervised transfer learning method is applied in a standard dataset CMATERDb 3.1.2[22-27]. The first layer worked as block called Conv1. It is a convolution layer that uses 64 filters with 7x7 kernel size followed by batch normalization and max pooling layer. Then comes common convolution layers with kernel size 3x3 but filter varies from 64, 128, 256 and 512. There are a block of six 3x3 Convolution layers with 64 filters. Next block contains eight 3x3 convolution layers with 128 filters, followed by another block that contains ten 3x3 convolution layers with 256filters. The last block consists of six 3x3 convolution layers with 512 filters. It ends up with a fully connected layer with 512 filters. In fully connected layer all node of the output layer is directly connected with all input layers node. This layer is responsible for classification based on the features extracted by its previous layers. It uses soft max activation function where as convolution and pooling layer uses Relu functions. Use of Relu non-

Table 3. Existing Work in Bengali Handwritten Character Recognition Domain

Methodology	Classifier and Data Set	Recognition Accuracy
(Rahman et al.[12], 2015)	CNN , 20000 Dataset,	85.36%
(Purkaystha et al[13], 2017)	CNN, 166105 Dataset	89.93%
(Hossain et al[14], 2018)	CNN, CMATERDb Dataset	95.01%
(Islam, et al.[15], 2018)	CNN, CMATERDb Dataset	95.71%
(Alom et al[16], 2018)	DCNN, 12000 Dataset	98.31%
(Rizvi et al[17], 2019)	SVM, RESNET 18, 277500 Dataset	98.04%
(Chatterjeeet al[18], 2020)	DCNN, Banglalekha Isolated	96.12%
(A. Roy[19], 2020)	Resnet 50, BanglaLekha Isolated	96.80%
(Das et al[20], 2021)	CNN , BanglaLekha Isolated	91.59%
(Khandokar et al[21], 2021)	CNN, NIST Dataset	92.91%

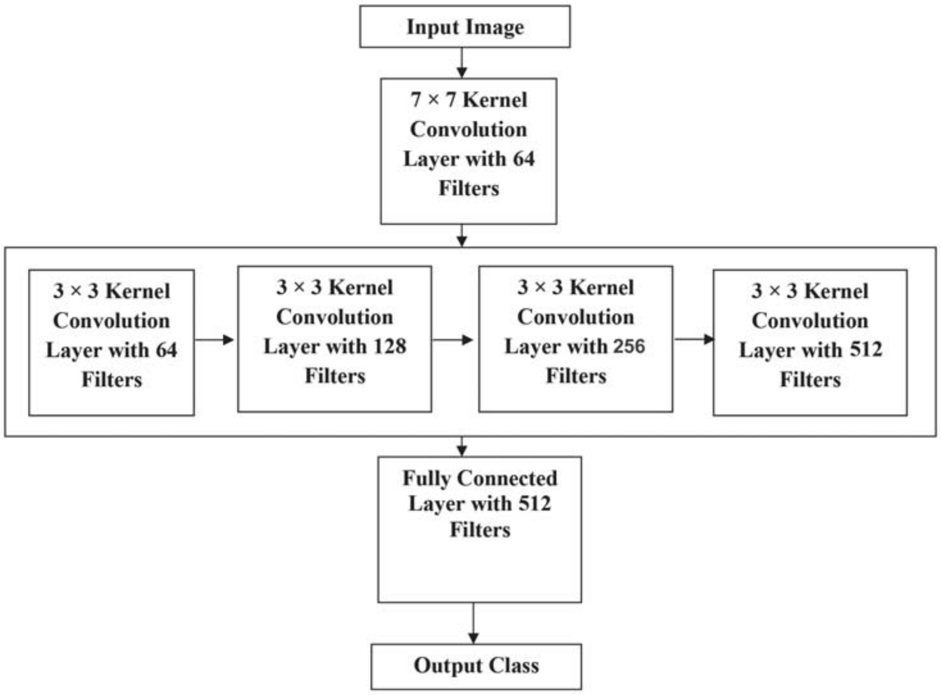

Fig.1. Supervised Model Architecture

linearity instead of Tanh or Sigmoid makes training faster. Each convolution block of the model is represented in fig.1 specifying their kernel size and number of filters.

Result and Discussion

The experiment is conducted in a system with Processor of AMD Ryzen 5 3550H (8 CPUs), ~2.1GHz , 16GB RAM , GPU: NVIDIA GeForce GTX 1650, GPU Memory: 4GB. First, we resized all the images in our dataset (both training and testing sets) to (120,120) to make them of uniform shape. The training set consisted of 50 classes, each of which had 240 images each making 240*50= 12000 images in our training data. The data set is divided into the training and validation set. 80% data set is used for train purpose which is around 9600 images and only 20% uses for testing data set which is around 2400 images. We first unfreezes the pre-trained layers of our model for training. After that we tried to estimate the best suitable learning rate for training our model as shown in Fig.2

After getting the suitable learning rate, we fine-tuned the model for 11 epochs. We used Gradient Clipping to have a stable training and prevent exploding gradients. It shows around 98.9% accuracy on the validation set after the completion of our training. It indicates that validation loss is always less than or almost equal to training loss. It indicates that our model is not suffering for under fitting as well as over fitting and accurately models the training data set. It shows that accuracy is saturated and loss becomes very stable after 800 batches.

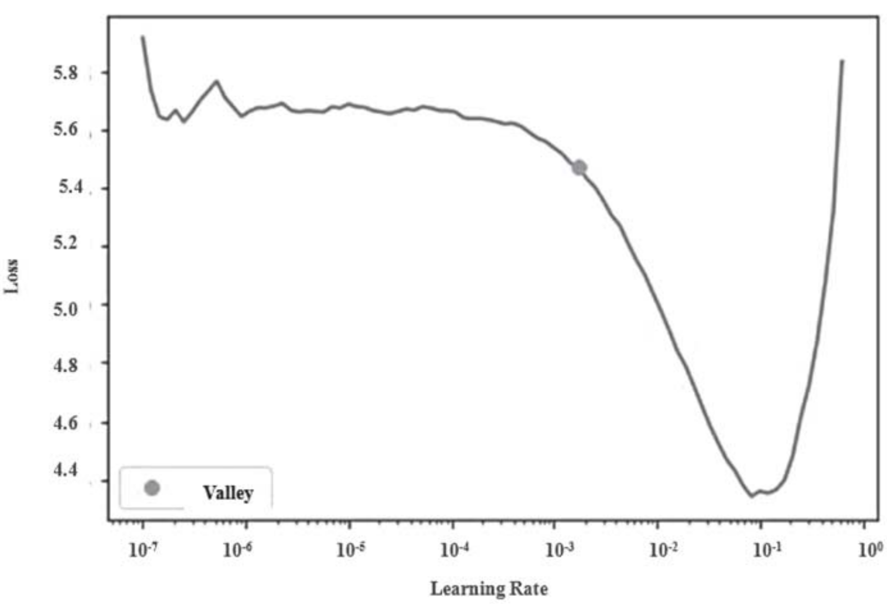

Fig.2. Learning Rate of the Model

Training loss is around 0.001 and validation loss is 0.053. Average epoch time is reduced to 1.04sec. Then it is applied upon 3000 test data set. It shows that 2969 test images are recognized accurately which gives overall result of 98.96% recognition accuracy on the new test data set. Corresponding confusion matrix is shown in Fig.4. 60 test sample images for each font class was taken. Diagonal elements of the matrix value indicates correct classification which shows that most of the font class is recognized correctly that is its value is 60 or very near to 60 in most of the cases.

Adam optimizer is used in training and loss functions are

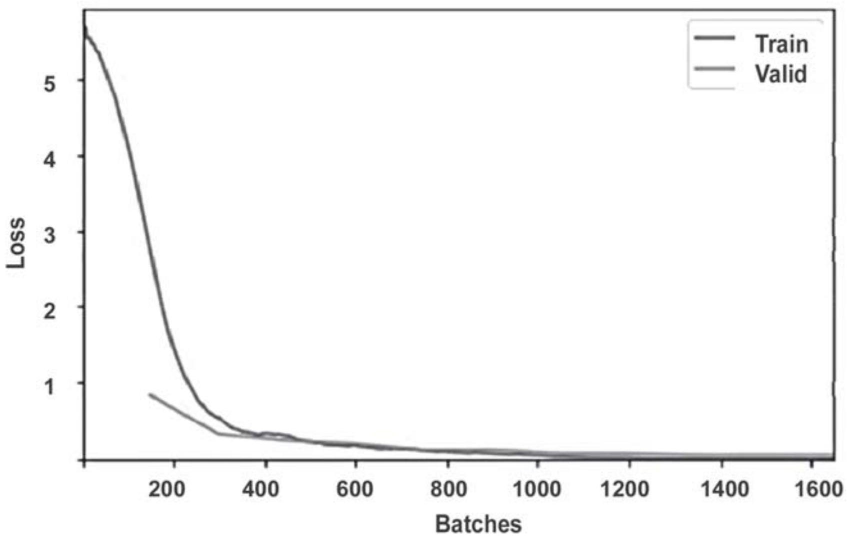

Fig.3. Training and Validation loss of the Model

Flattened Loss of Cross Entropy Loss. Performance of this supervised approach for each individual font class is compared in the Fig.5. There are 50 font classes. For each font classes, 60 samples are used for testing. Average font classification accuracy of vowel fonts is around 99.24% whereas consonant font classification accuracy is around 98.9%. Overall accuracy is 98.96%. Out of 50 font classes 28 font classes (like অ, আ) is classified with 100% accuracy. Eight vowels (অ, আ, ঈ, উ, ঊ, ঋ, ঐ ও) out of 11 has classified with 100% accuracy. Only three font classes (খ, ন, স) have achieved low

is 60 or very near to 60 in most of the cases.

	অ	আ	ই	ঈ	উ	ঊ	ঋ	এ	ঐ	ও	ঔ	ক	খ	গ	ঘ	ঙ	চ	ছ	জ	ঝ	ঞ	ট	ঠ	ড	ঢ	ণ	ত	থ	দ	ধ	ন	প	ফ	ব	ভ	ম	য	র	ল	শ	ষ	স	হ	ড়	ঢ়	য়	ৎ	ং	ঃ	ঁ
অ	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
আ	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ই	0	0	58	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
ঈ	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
উ	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঊ	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঋ	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
এ	0	0	0	0	0	0	0	58	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঐ	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ও	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঔ	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ক	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
খ	0	0	0	0	0	0	0	0	0	0	0	0	57	0	1	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
গ	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঘ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঙ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
চ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ছ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
জ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঝ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঞ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ট	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঠ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ড	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ঢ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ণ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
ত	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
থ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
দ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ধ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ন	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	57	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
প	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ফ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ব	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ভ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
ম	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0	0	0	0	0
য	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0	0	0
র	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	1	0	0	0	0
ল	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0	0	0	0	0
শ	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0	0
ষ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	59	0	0	0	0	0	0	0	0	0
স	1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	57	0	0	0	0	0	0	0	0
হ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0	0
ড়	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0	0
ঢ়	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0	0
য়	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0	0
ৎ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0	0
ং	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0	0
ঃ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	60	0
ঁ	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	58

Fig.4 Confusion Matrix

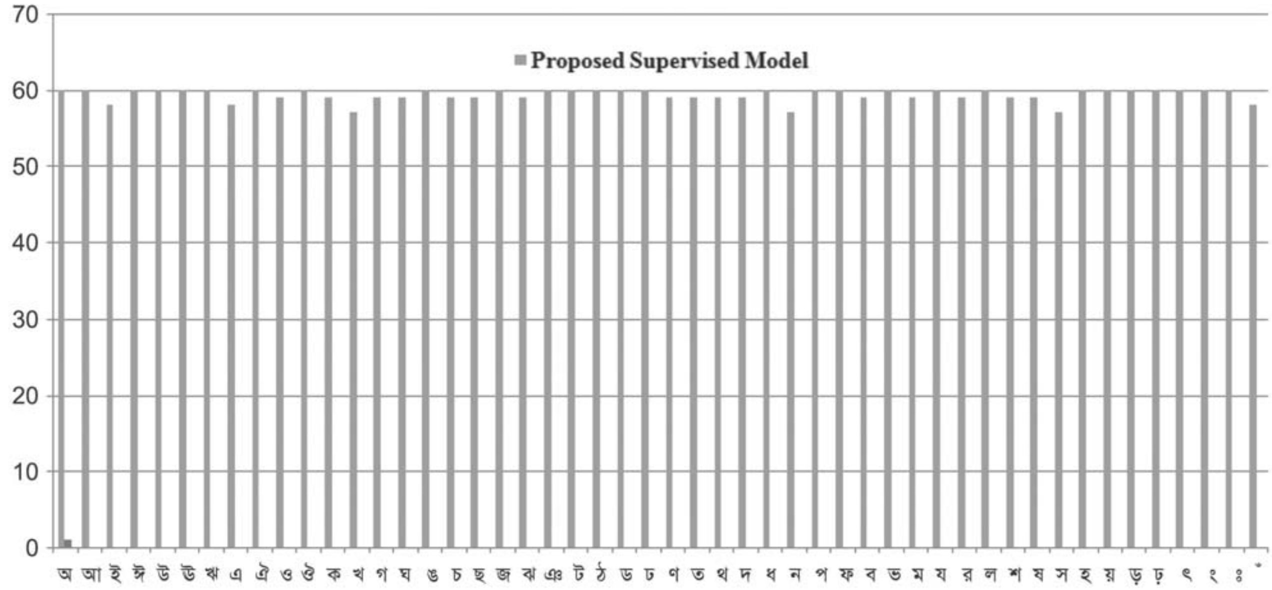

Fig.5. Classification of 50 Different Basic Font Class

accuracy of 95% in our used dataset CMATERD b 3.1.2.

Experimental results are compared with some existing recent research work in this domain in Fig.6. It shows that proposed method performs best upon all of the previous existing work. Most of research work uses their own data set or features are extracted and then passed to neural network or support vector machine or any other multi layer perception based classifier. Our approach does not need any preprocessing and separate feature extraction process. Average accuracy of Bengali Handwritten character recognition varies from 87% to 98%. Our proposed model shows best classification accuracy 98.96% achieved so far on Bengali basic handwritten character recognition approach. In comparison with other CNN models it needs less number of parameters around 21M.


(a) (b) (c) (d)

Fig.7. Images Effecting More in Reducing Recognition Rate

Images that are contributing to higher loss are shown in the Fig.7. Images are highly distorted or misclassified labelling is the main reason of the recognition error or loss. Image (a) and (b) are labeled wrongly while (c) suffers from overlapping or overwriting problems and (d) is an example of distorted image that even human eyes also can't recognize.

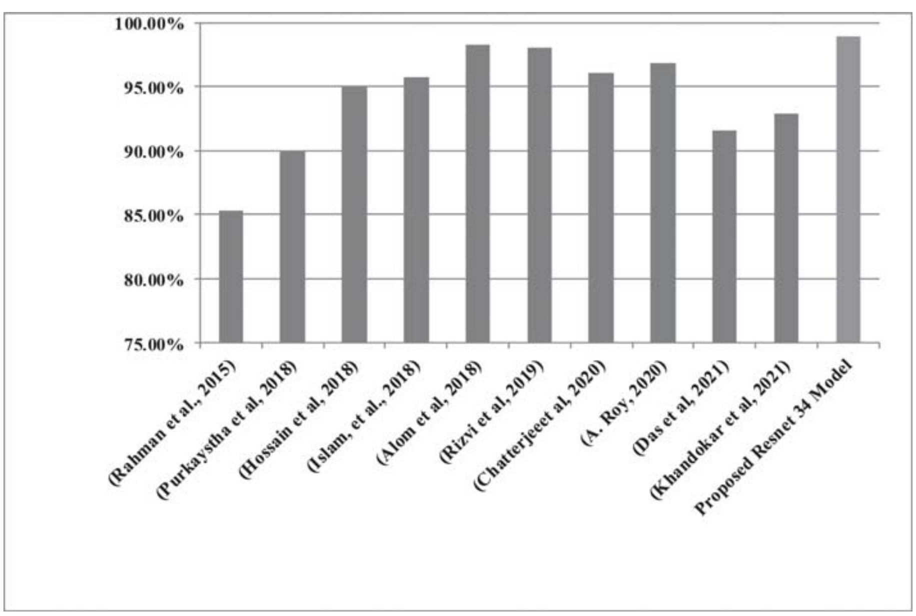

Fig.6. Test Results Comparison with Existing Work

Conclusion

Deep learning using various types of convolution neural network became very popular and well accepted model in handwriting character recognition in last one decade because of its high tolerance of noise and less preprocessing requirements. Our proposed model is showing best accuracy of 98.96% over a standard dataset. Initially it is tested upon CMATERDb 3.1.2 data set. Other dataset like Bangla-lekha, ISI will also be merged in our dataset to increase number of sample and to check the result in crossed dataset platform. Increasing the number of sample images will help in training that will enhance the performance.

Maximum compound characters and modifiers are untouched or very less work has been done. This work may be further extended in this direction. In future online dataset will be tested or model will be designed that can be applied directly to the word level recognition. The output of OCR can be converted to speech that can help visually handicapped persons. ❐

References

1. B.B. Chaudhuri, and U. Pal, *Pattern recognition*, ***31*** (5) , 531-549, (1998).
2. S. Hossain, N. Akter, H. Sarwar, C. M. Rahman, and M. Mori, *Character Recognition*, **9,** 83-112, (2010).
3. U. Pal, and B.B. Chaudhuri, *Pattern Recognition*, **37** (9), 1887-1899, (2004).
4. S.D. Connell, and A. K. Jain, *Pattern Recognition*, **34** (1), 1-14, (2001).
5. Swapan Parui, K. Guin, Ujjwal Bhattacharya, and Bidyut Chaudhuri, *Neural Computing and Applications*, **32,** No. 14 , 1 – 4, (2009).
6. P.P. Roy, P. Dey, S. Roy, U. Pal and F. Kimura, 14th International Conference on Frontiers in Handwriting Recognition, pp. 661-666, (2014).
7. P. P. Roy, A. K. Bhunia, A. Das, P. Dey, and U. Pal, *Pattern Recognition*, **60**, 1057-1075. (2016).
8. T.K. Bhowmik, P. Ghanty, , A. Roy, and S. K. Parui, *International Journal on Document Analysis and Recognition (IJDAR)*, **12** (2), 97-108, (2009).
9. M.R. Kibria, A. Ahmed, Z. Firdawsi and M.A. Yousuf, IEEE Region 10 Symposium (TENSYMP), 2020, pp. 965-968, (2020).
10. B. Kannan and P.J. Jino, *Dyuthi*, (2014).
11. B. Rajyagor, and R. Rakhlia, *International Journal of Recent Technology and Engineering (IJRTE)*, **8 (6),** 5815-5819, (2020).
12. M. M. Rahman, M. A. H. Akhand,S. Islam, P. Chandra Shill, and M. M. Hafizur Rahman, International Journal of Image, *Graphics and Signal Processing*, **7 (8),** 42–49, (2015).
13. B. Purkaystha, T. Datta, and M. S. Islam, 20th International Conference of Computer and Information Technology, ICCIT 2017, 1–5, (2017).
14. Akm Shahariar Azad Rabby y, S. Haque, S. Abujar, and S. A. Hossain, *Procedia Computer Science*, **143**, 603–610, (2018).
15. Akm Shahariar Azad Rabby , Sadeka Haque, Sanzidul Islam, Sheikh Abujar, and Syed Akhter Hossain , *Procedia Computer Science*, **143**, 528–535, (2018).
16. M.A.I. Rizvi, K. Deb, M. I. Khan,M. M. M. S. Kowsar and T. Khanam, *Turkish Journal of Electrical Engineering and Computer Sciences*, **27**(4), 3195–3207, (2019).
17. T.M.T. Md Zahangir Alom , Paheding Sidike , Mahmudul Hasan and V. K. Asari, *Comput. Intell. Neurosci.*, **2018**, no. Article ID 6747098, p. 13 Pages, (2018).
18. S. Chatterjee, R. K. Dutta, D. Ganguly, K. Chatterjee and S. Roy, Intelligent Human Computer Interaction. IHCI 2019, 11886(Springer, Cham), (2020).
19. A. Roy, *International Journal of Recent Technology and Engineering*, **8** (6), 5815–5819, (2020).
20. T.R. Das, S. Hasan,M. R. Jani, F. Tabassum, and M. I. Islam, *Journal of Computer and Communications*, **09**(03), 158–171, (2021).
21. I. Khandokar, M. Hasan, F. Ernawan, S. Islam, and M. N. Kabir, *Journal of Physics: Conference Series*, **1918** (4), 593–611, (2021).
22. N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D.K. Basu, Applied Soft Computing, **12**, pp. 1592-1606, (2012).
23. N. Das, J.M. Reddy, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D.K. Basu, *Applied Soft Computing*, **12**, pp. 2486-2495, (2012).
24. N. Das, K. Acharya, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, Proceedings of the International Conference on Information Systems Design and Intelligent Applications 2012 (INDIA 2012) held in Visakhapatnam, India, January 2012. **132**, pp. 145-152, (2012).
25. N. Das, S. Basu, R. Sarkar, M. Kundu, M. Nasipuri, and D.K. Basu, in 4th Indian International Conference on Artificial Intelligence, Bangalore, 2009, pp. 1901-1913, (2009).
26. N. Das, S. Basu, R. Sarkar, M. Kundu, M. Nasipuri, and D.K. Basu, in International conference on Signal and Image Processing, Mysore, India, 2009, pp. 451-454, (2009).
27. N. Das, K. Acharya, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri, *IJDAR*, **17,** No. 4, (2014).