IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 14, No. 5, October 2025, pp. 4171∼4180
ISSN: 2252-8938, DOI: 10.11591/ijai.v14.i5.pp4171-4180 ❒ 4171

BonoNet: a deep convolutional neural network for recognizing bangla compound characters

Kazi Rifat Ahmed[1], Nusrat Jahan[2,3], Adiba Masud[1,4], Nusrat Tasnim[5], Sazia Sharmin[6], Nusrat Jahan Mim[1], Imran Mahmud[1]

[1]Department of Software Engineering, Faculty of Science and Information Technology, Daffodil International University, Dhaka, Bangladesh
[2]Department of Information Technology and Management, Faculty of Science and Information Technology, Daffodil International University, Dhaka, Bangladesh
[3]Faculty of Electronic Engineering and Technology (FKTEN), Universiti Malaysia Perlis, Arau, Malaysia
[4]Department of Computer Science, College of AI, Cyber and Computing, University of Texas at San Antonio, San Antonio, United States
[5]Department of Information and Communication Technology, Bangladesh University of Professionals, Dhaka, Bangladesh
[6]Department of Computer Science, Faculty of Science and Technology, American International University, Dhaka, Bangladesh

Article Info

Article history:

Received Aug 11, 2024
Revised Jun 28, 2025
Accepted Aug 6, 2025

Keywords:

Bangla
BonoNet
Compound characters
Deep convolutional neural network
Handwritten
Optical character recognition

ABSTRACT

The bangla alphabet includes vowels, consonants, and compound symbols. The compound nature of bangla is a product of combining two or more root bangla characters into one graph. They are difficult to differentiate because they have a sophisticated geometric shape and an immense variety of scripts used by different places and individuals. This is one of the greatest challenges in creating effective optical character recognition (OCR) systems for bangla. In this paper, a deep convolutional neural network (DCNN)-based system is presented to identify bangla compound characters with high precision. The model was trained using the AIBangla dataset. It has about 171 classes of bangla compound characters. A DCNN system, BonoNet, was designed to classify compound characters. BonoNet outperformed all other state-of-the-art architecture on the test set and improved over current state-of-the-art architecture methods. BonoNet will greatly improve the automation and analysis of the bangla language by accurately identifying these compound complex characters.

Corresponding Author:

Nusrat Jahan
Department of Information Technology and Management, Faculty of Science and Information Technology
Daffodil International University
Dhaka, Bangladesh
Email: nusrat.swe@diu.edu.bd

1. INTRODUCTION

Bangla is the seventh most widely spoken language on earth, and it is spoken by nearly 300 million people in South Asia's Bengali region. It is the official and national language of Bangladesh and is spoken by close to 98% of Bangladesh's population. The script of the language has vowels, consonants, and complex letters within distinctive and separate visual structures, leading to a distinctive and elaborate system of writing. This complexity has proven to be difficult for optical character recognition (OCR) to handle, especially in the case of character recognition of handwritten documents on physical media. OCR technology has been prized as an invaluable resource for digitizing written materials for many years, but bangla's compound characters pose unique challenges according to their structural complexity and diversity.

Various computational solutions exist to be implemented in handwritten character recognition. These include machine learning techniques, artificial neural networks (ANN), multilayer perceptrons (MLP), support vector machines (SVM), and growing emphasis on deep learning models such as convolutional neural networks (CNNs) [1], [2]. The CNNs themselves have long been in demand for having greater precision and less reliance on human-created feature extraction. Their ability to learn visual features automatically at a hierarchical level makes them extremely resourceful in image recognition and classification tasks.

Bangla script layout consists of 50 simple characters, out of which 11 are vowels and 39 are consonants [3]. They together create more than 171 compound characters by the combination of simple ones. Despite remarkable advancements in studies, the majority of the earlier research mostly had an aim to recognize simple characters only. Compound characters, due to their unpredictability and strong variability, have been less explored in existing OCR models [4]. There thus remains an enormous scope for systems capable of handling such complexity with robust accuracy and generalizability. Figure 1 shows examples of the simple and compound characters used in this work.

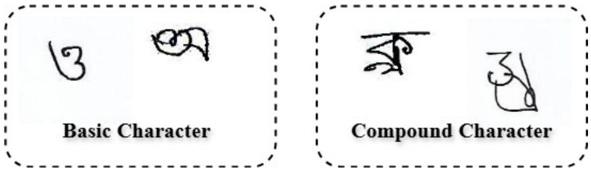


Figure 1. Bangla basic and compound characters example

In the recent past, several deep learning-inspired models have been introduced for bangla character recognition. Ahmed *et al.* [5] introduced a deep convolutional neural network (DCNN) with 76,000 training images for character classification, while Ashiquzzaman *et al.* [6] employed exponential linear unit (ELU)-based methods to enhance performance on the CMATERDB 3.1.3.3 dataset. Azad *et al.* [7] introduced DConvAENNet, an autoencoder-DCNN combination, on datasets such as BanglaLekha-Isolate and Ekush. Uddin *et al.* [8] used hybrid ConvLSTM to show good performance on identifying bangla handwritten digits. Begum *et al.* [9] used longest run (LR)+chain code histogram (CH) features, whereas Chakraborty and Paul [10] did bidirectional conversion from simple to compound and vice versa. Chowdhury *et al.* [11] achieved improved accuracy using CNN with data augmentation, whereas Hasan *et al.* [12], [13] experimented with VGG-16, ResNet-50, and DenseNet—identifying DenseNet particularly effective for simple as well as compound characters on the AIbangla dataset.

Other approaches followed handcrafted features and combination strategies. Kibria *et al.* [14] employed SVM and MLP classifiers with local receptive field (LRF), histogram of oriented gradients (HOG), and diagonal features, and Khan *et al.* [15] followed high performance on the BanglaLekha-Exclusive dataset using SE-ResNeXt. Mukherjee *et al.* [16] experimented with various learning methods on 10,000 bangla web images. Saha *et al.* [17], [18] introduced BBCNet-15 for improved basic character recognition and compared local binary pattern (LBP)-based descriptors under various classes. Sarika *et al.* [19] demonstrated VGG-16 performance for Telugu script, and Rabi *et al.* [20] demonstrated excellent results for KDANet on BanglaLekha. Pramanik and Bag [21] used chain-code features for compound character recognition using ICDAR and CMATERdb databases. Koiso *et al.* [22] extended OCR research to Japanese script. Separately, Jishan *et al.* [1] integrated NLP with hybrid neural networks for text image recognition, utilizing grammar analysis and language modeling techniques, along with other researchers, used NLP to recognize different channels from images and texts [23]–[25].

Despite such a heterogeneous body of work, most works still emphasize unique character recognition. Handwritten compound characters are still difficult to classify since they are visually variable and contextually variable. To address this issue, in this work, a shallow DCNN architecture named BonoNet is introduced that is specifically targeted towards the accurate recognition of bangla compound characters. BonoNet outperforms the state-of-the-art models ResNet and DenseNet on the AIBangla dataset. Unlike other methods, BonoNet automates feature extraction and tackles the high intra-class similarity and inter-class variability common in compound bangla characters.

2. METHOD

In this section, the approach of selecting bangla compound characters which will be utilized is explained. A skilled DCNN, 'BonoNet', has been designed to identify bangla compound characters efficiently. The suggested approach is illustrated in Figure 2.

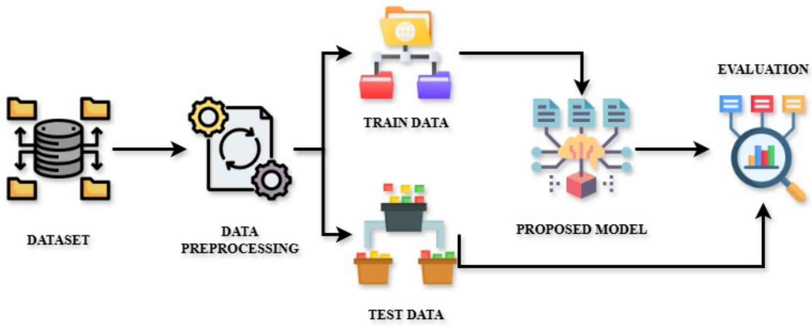


Figure 2. Proposed methodology for compound character recognition

2.1. Data collection

The methodology that was suggested was adapted from the AIBangla dataset created by Hasan *et al*. [12]. The dataset consists of handwritten bangla characters through submissions of more than 2,000 individuals from various institutes in Bangladesh. It is a new benchmark in the field with a holistic use case and performance benchmark. AIBangla dataset has a large bangla character set, including 249,911 images for compound characters and 80,403 images for simple characters in 50 classes. As their dataset does not contain any number data, AIBangla gathered 330,314 images for 221 classes. They also have a dataset of 171 compound characters in bangla, which we will use. The AIBangla dataset samples are represented in Figure 3.

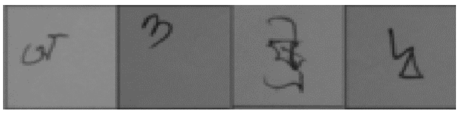


Figure 3. Few examples of the dataset

2.2. Data preprocessing

Data processing is beneficial for improving accuracy and reducing the complexity of an image. Python OpenCV was utilized to implement the preprocessing steps. Figure 4 shows the preprocessing steps that were adopted here. For achieving that, it was initiated by transforming RGB images into gray-scale to lower their dimensionality and mitigate the load on the model. The dataset needs to be transformed to gray-scale so as to cancel tone variability and noise. OpenCV library in Python is utilized for achieving that, which succeeds in transforming images into gray-scale and removing noise by making use of Gaussian blur. Image thresholding also reduces analysis by transforming images into binary black and white. Particularly utilized multi-otsu thresholding, which classifies pixels into classes as per their intensity of gray level. With the use of OpenCV under Python, multi-otsu thresholding is used over the dataset, enhancing its preparatory procedure. To find precise handwritten compound characters, the unnecessary parts of the image are eliminated. Utilizing contour detection from Python's OpenCV, to detect the edges of Bangla compound characters. After detection, the image is cropped to the size of the character. Image resizing is another critical step, which accelerates neural network training by minimizing the pixels. In our instance, images are resized to 28×28, leading to better model results as presented in Figure 5. The results before pre-processing are shown in Figure 5(a) and after pre-processing are shown in Figure 5(b).

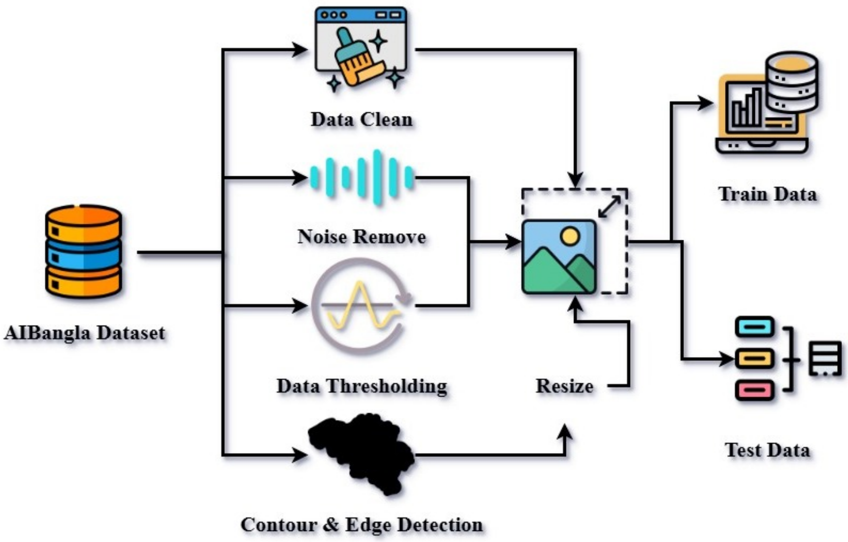


Figure 4. Data preprocessing steps of train and test data

(a) (b)

Figure 5. The dataset of (a) before preprocessing and (b) after preprocessing

Training, test, and validation sets have been divided. To be specific, 80% of the data were devoted to training purposes, and 10% to test purposes, with the remaining 10% for validation purposes. Complex characters are the center of attention, which are further classified into 171 categories. A total of 199,803 samples are present for utilization in training the model. A total of 25,123 samples are also present for testing the model's performance. Additionally, 24,908 samples are present for validating the performance of the model while training it. The model can be exhaustively trained, tested, and validated with this provision.

2.3. Proposed method: BonoNet architecture

This DCNN produces 28×28 images with 7 convolutional layers, 3 fully connected layers, and 5 dropout layers. The first and second layers: kernel of size 3×3 and 32 filters. In the model, a batch-normalization layer with pool size of 22 and without strides value is used after each convolution layer. Rectified linear unit (ReLU) is an activation function for all convolutional layers, and max pooling is skipped by the first layer. For reducing overfitting, a dropout layer is applied after the second layer, and then a max pooling layer. All third and fourth convolutional layers consist of 64 filters with an activation function. ReLU activation function and a maxpooling layer were applied after the fourth layer. Batch normalization is applied everywhere except the output layer. A dropout layer after the max pool layer is complete. 5^{th}, 6^{th}, and 7^{th} have 128, 128, and 256 filters. BatchNormalization followed by max pooling and dropout in each layer. Then we have the flatten layer to classify, followed by three fully connected layers of 512, 512, and 171 neurons. Batchnormalization layer on the last layer, dropout in the lower two. Uses softmax activation in the final layer. Figure 6 is the proposed DCNN model.

2.4. Model breakdown

The BonoNet architecture is organized into three main components. It consists of a feature extractor and a classifier. It also includes training parameters that optimize the overall performance of the model.

2.4.1. Feature extractor

Feature extraction lies at the foundation of the BonoNet architecture, dealing with input image data efficiently. Raw inputs are converted into structured features through the application of several layers. Processing initiates with an input layer, with the convoy layers for recognizing the features.

- Input layer: the model begins with an input layer that takes in the image data.
- Convolutional layers (Conv2D): multiple convolutional layers extract features from the image. Each layer applies filters to detect patterns like edges, textures, and shapes at different levels of abstraction.
- ReLU activation: ReLU is used to introduce non-linearity, helping the model capture complex patterns.
- Batch normalization: this layer normalizes the output of the previous layer.
- Max pooling: max pooling downsamples the feature maps, reducing dimensionality and computational cost while preserving important information.
- Dropout: dropout randomly deactivates a portion of neurons during training, preventing overfitting.

2.4.2. Classifier

Dense layers are fully connected layers that use the extracted features to determine the class of the image. The architecture includes an initial dense layer with 512 units, followed by a layer with 1,024 units. Finally, there is an output layer with a number of units equal to the number of classes.

2.4.3. Model training parameter

The proposed model was trained using a set of designated parameters optimized to ensure effective convergence and generalization. Table 1 shows the parameters has been used in the proposed methodology. These parameters were carefully selected to enhance the model's training performance.

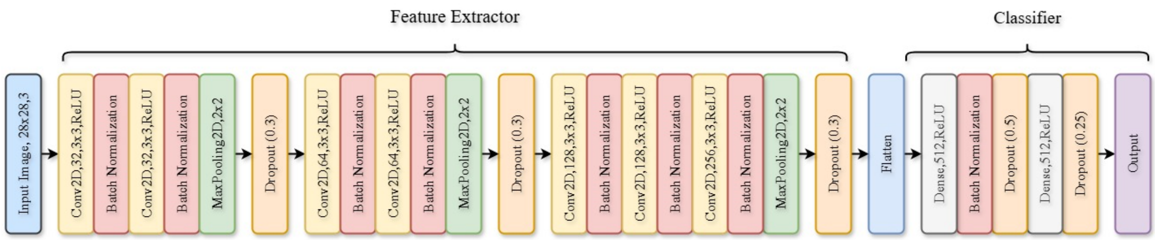


Figure 6. BonoNet architecture

Table 1. Training parameters used for BonoNet

Parameter	Value
Learning rate	0.0001
Decay factor	0.2
Early stopping patience	2
Loss function	Categorical cross-entropy
Total epochs	100
Epochs before stopping	36
Evaluation dataset	Fresh validation set

2.5. Benefits for image processing

The structure of a CNN is exactly what is needed to solve image processing problems because it is possible to train it to learn hierarchical representations of visual features. The convolutional layers are able to detect local features, and the pooling layers offer dimensionality reduction and translation invariance. The dense layers then concatenate these features ultimately to enable accurate prediction. The use of batch normalization and dropout methods improves training efficiency and generalization and thus makes the model invariant to image data variation.

3. RESULTS AND DISCUSSION

3.1. Setup and environment

The research made use of the mentioned resources and specifications. The experiment was conducted by using a CPU model Intel(R) Core (TM) i5-8265U running at 1.60 GHz and 1.80 GHz with 8 GB of RAM. The device is equipped with 8 GB of RAM, Intel(R) UHD Graphics 620 and NVIDIA GeForce MX110 GPU, Toshiba MQ04ABF100 HDD, and Windows 11 Home operating system. The program was executed using Jupyter Notebook 6. 4.12 as well as the Anaconda platform version 2.3.

3.2. Experiment results

The 'BonoNet' model was tested with 10% of images from the datasets. From the 171 classes of image classification, this model acquired around 90.01% training accuracy and 89.99% validation accuracy. This model also obtained 90.01% precision in the train and 90.01% in the validation set. Overall, it provided 90.01% accuracy in recognizing bangla compound characters. Table 2 shows the result achieved in classification.

The evaluation criteria for the model depend on its performance with the validation dataset, which contains 24,908 samples. The model's precision, recall, and F1 score are all 0.90, indicating its 90% accuracy in correctly identifying the class in its predictions (precision), in the actual instances present (recall), and in overall performance considering both precision and recall (F1 score). This shows that the model is reliable and effective in accurately predicting the correct classes for the validation data. Figure 7 shows the training and validation accuracy of the 'BonoNet' model for identifying the Bangla compound characters.

Table 2. Detailed metrics of BonoNet model

Class	Precision	Recall	F1-score	Support
class 0	0.89	0.89	0.89	1169
class 1	0.90	0.90	0.90	1170
...	...	...	...	....
class 170	0.89	0.89	0.89	1169
micro avg	0.89	0.90	0.90	199803
macro avg	0.90	0.89	0.89	199803
weighted avg	0.90	0.90	0.90	199803

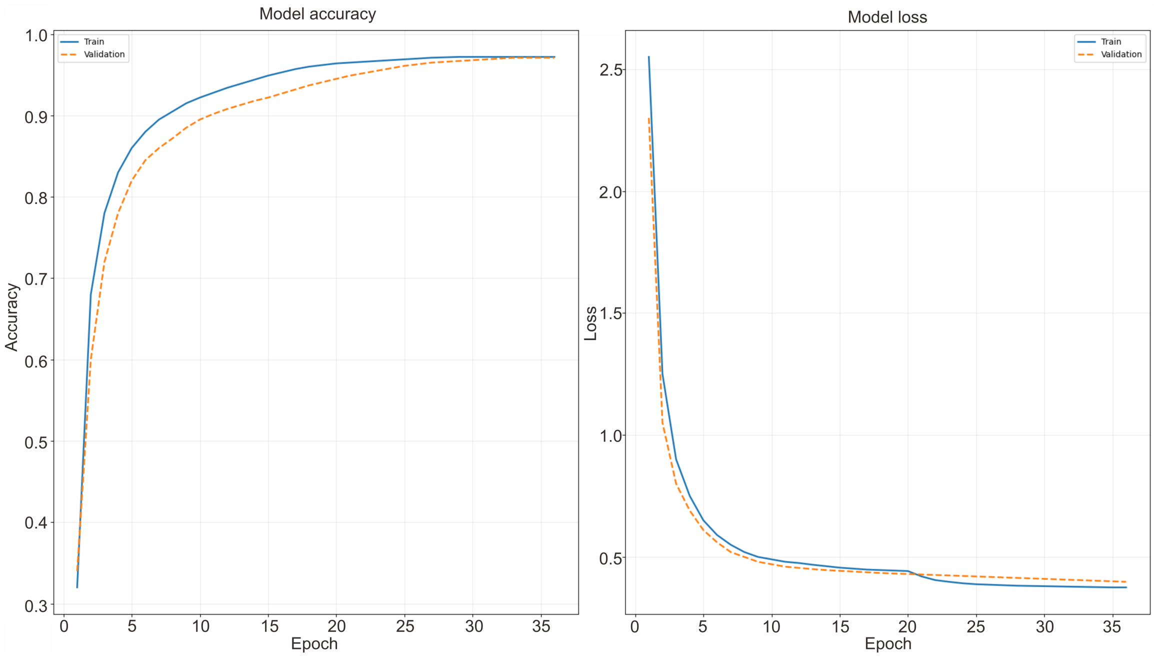


Figure 7. BonoNet model training and validation accuracy and loss

The charts display how well the model performed during 35 epochs. The model's accuracy increases as it trains, as shown in the left plot. Both training and validation accuracy initially begin at a low level and quickly improve, reaching a plateau of about 90%, indicating improved predictive abilities of the model. The model's loss, which is a measure of error, is displayed on the plot to the right. The beginning and final states for both training and validation loss start quite high and proceed to drop off significantly until both levels stabilize at a low level. This shows that the model is learning effectively, with errors decreasing as it undergoes training. In general, the data indicates that the model is performing uniformly well on both the training and validation sets, achieving high accuracy with minimal errors. The 'BonoNet' model successfully classified compound characters with intricate structures, leading to lower errors compared to simple and numeral characters. Table 3. displays the comparison between the accuracy and class categorization of the 'BonoNet' model and the current model.

Table 3. Comparison with existing and proposed models

Name	Number of classes/images	Accuracy (%)
Chakraborty and Paul [10]	300,000	89.20
Hasan *et al.* [12]	171	81.83
Kibria *et al.* [14]	171	85.91
Pramanik and Bag [21]	171	88.74
Saha *et al.* [17]	171	73.3
Proposed Model (BonoNet)	171	90.01

Here, Table 3 illustrates the precision of various models in identifying bangla compound characters. In their study, Chakraborty and Paul [10] obtained an 89.20% accuracy using a vast dataset containing 300,000 images. Hasan *et al.* [12], Kibria *et al.* [14], Pramanik and Bag [21], and Saha *et al.* [17] utilized datasets comprising 171 classes and attained accuracies of 81.83%, 85.91%, 88.74%, and 73.3% in the same order. Similarly, the BonoNet model, which was proposed, utilized a dataset consisting of 171 categories and managed to reach an accuracy rate of 90.01%, the highest among all models. This indicates that the BonoNet model is more accurate than the other models for this particular task. The 'BonoNet' model surpasses various models to achieve improved results in recognizing compound characters.

4. CONCLUSION

Recently, CNN has gained much notice due to its advanced ability to categorize images effectively. The model is consistently relevant. The 'BonoNet' model, developed with CNN, outperformed the prior model in accurately recognizing bangla compound characters. We utilized the model to improve the results. The model 'BonoNet' can achieve optimal recognition accuracy for accurate identification of bangla compound characters. Conclusions were compared to the graphs generated to verify the model. Graphs were produced for accuracy and loss functions at each cycle. The proposed models achieved a 90.01% level of accuracy. To enhance the model's accuracy for potential growth in the future. Using more advanced and operational devices in addition to our trained device can enhance the accuracy of the proposed model. The training potential of the dataset will grow as time is saved. Increasing the size of the datasets for training and validation could potentially improve the odds. Alternatively, the model can be trained using larger image input sizes in order to potentially improve results. Only individual bangla compound characters can be used with the suggested method. In the future, we aim to combine simple and complex bangla characters to define a complete bangla word within a sentence.

ACKNOWLEDGMENTS

We would like to thanks all the authors for their contribution. The Daffodil International University has provided a great support for providing the environment to do the research.

FUNDING INFORMATION

No financial support was received for the completion of this study.

AUTHOR CONTRIBUTIONS STATEMENT

This journal uses the Contributor Roles Taxonomy (CRediT) to recognize individual author contributions, reduce authorship disputes, and facilitate collaboration.

Name of Author	C	M	So	Va	Fo	I	R	D	O	E	Vi	Su	P	Fu
Kazi Rifat Ahmed	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Nusrat Jahan			✓			✓				✓	✓	✓		
Adiba Masud					✓			✓		✓	✓		✓	✓
Nusrat Tasnim			✓			✓		✓		✓			✓	✓
Sazia Sharmin		✓					✓	✓		✓			✓	✓
Nusrat Jahan Mim				✓		✓		✓		✓			✓	✓
Imran Mahmud	✓		✓							✓		✓		

C : **C**onceptualization
M : **M**ethodology
So : **So**ftware
Va : **Va**lidation
Fo : **Fo**rmal Analysis
I : **I**nvestigation
R : **R**esources
D : **D**ata Curation
O : Writing - **O**riginal Draft
E : Writing - Review & **E**diting
Vi : **Vi**sualization
Su : **Su**pervision
P : **P**roject Administration
Fu : **Fu**nding Acquisition

CONFLICT OF INTEREST STATEMENT

The authors affirm that this study was conducted without any conflicting interests.

DATA AVAILABILITY

The data supporting this research are directly available on Kaggle via https://www.kaggle.com/datasets/awmium/handwritten-bangla-characterdataset-aaibangla, originally published by the dataset authors in association with the paper available at https://doi.org/10.1109/ICBSLP47725.2019.201481. The dataset, titled "Handwritten bangla character dataset (AI-Bangla)", was used under the terms specified by its public release.

REFERENCES

[1] M. A. Jishan, K. R. Mahmud, A. K. Al Azad, M. R. A. Rashid, B. Paul, and M. S. Alam, "Bangla language textual image description by hybrid neural network model," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 2, pp. 757–767, 2020, doi: 10.11591/ijeecs.v21.i2.pp757-767.

[2] M. G. Hussain, B. Sultana, M. Rahman, and M. R. Hasan, "Comparison analysis of Bangla news articles classification using support vector machine and logistic regression," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 21, no. 3, pp. 584–591, 2023, doi: 10.12928/TELKOMNIKA.v21i3.23416.

[3] A. Hasan, M. H. Jobayer, M. A. A. M. Pias, T. Alam, and R. Khan, "Bangla sign language recognition with multimodal deep learning fusion," *Engineering Reports*, vol. 7, no. 4, 2025, doi: 10.1002/eng2.70139.

[4] M. Kabir, O. B. Mahfuz, S. R. Raiyan, H. Mahmud, and M. K. Hasan, "BanglaBook: A large-scale bangla dataset for sentiment analysis from book reviews," *arXiv-Computer Science*, 2023, doi: 10.48550/arXiv.2305.06595.

[5] S. Ahmed, F. Tabsun, A. S. Reyadh, A. I. Shaafi, and F. M. Shah, "Bengali handwritten alphabet recognition using deep convolutional neural network," *5th International Conference on Computer, Communication, Chemical, Materials and Electronic Engineering (IC4ME2)*, 2019, doi: 10.1109/IC4ME247184.2019.9036572.

[6] A. Ashiquzzaman, A. K. Tushar, S. Dutta, and F. Mohsin, "An efficient method for improving classification accuracy of handwritten Bangla compound characters using DCNN with dropout and ELU," *2017 3rd IEEE International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, pp. 147–152, 2017, doi: 10.1109/ICRCICN.2017.8234497.

[7] M. A. Azad, H. S. Singha, and M. M. H. Nahid, "Bangla handwritten character recognition using deep convolutional autoencoder neural network," *2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT)*, 2020, doi: 10.1109/ICAICT51780.2020.9333472.

[8] A. H. Uddin, J. Khatun, M. A. Meghna, and P. Mahmud, "Bangla handwritten digit recognition using RNN-CNN hybrid approach," *2022 25th International Conference on Computer and Information Technology (ICCIT)*, pp. 288–293, 2022, doi: 10.1109/ICCIT57492.2022.10055089.

[9] H. Begum, A. Rafid, and M. M. Islam, "Recognition of bangla handwritten characters using feature combinations," *2018 5th IEEE Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, 2018, doi: 10.1109/UPCON.2018.8597076.

[10] S. Chakraborty and S. Paul, "Bengali handwritten character transformation: basic to compound and compound to basic using convolutional neural network," *International Conference on Robotics, Electrical and Signal Processing Techniques*, pp. 142–146, 2021, doi: 10.1109/ICREST51555.2021.9331247.

[11] R. R. Chowdhury, M. S. Hossain, R. U. Islam, K. Andersson, and S. Hossain, "Bangla handwritten character recognition using convolutional neural network with data augmentation," *2019 Joint 8th International Conference on Informatics, Electronics and Vision (ICIEV) and 2019 3rd International Conference on Imaging, Vision and Pattern Recognition (icIVPR)*, pp. 318–323, 2019, doi: 10.1109/ICIEV.2019.8858545.

[12] M. M. Hasan, M. M. Abir, M. Ibrahim, M. Sayem, and S. Abdullah, "AIBangla: A benchmark dataset for isolated bangla handwritten basic and compound character recognition," *2019 International Conference on Bangla Speech and Language Processing (ICBSLP)*, 2019, doi: 10.1109/ICBSLP47725.2019.201481.

[13] M. N. Hasan, R. I. Sultan, and M. Kasedullah, "An automated system for recognizing isolated handwritten bangla characters using deep convolutional neural network," *ISCAIE 2021 - IEEE 11th Symposium on Computer Applications and Industrial Electronics (ISCAIE)*, pp. 13–18, 2021, doi: 10.1109/ISCAIE51753.2021.9431799.

[14] M. R. Kibria, A. Ahmed, Z. Firdawsi, and M. A. Yousuf, "Bangla compound character recognition using support vector machine (SVM) on advanced feature sets," *2020 IEEE Region 10 Symposium (TENSYMP)*, pp. 965–968, 2020, doi: 10.1109/TENSYMP50017.2020.9230609.

[15] M. M. Khan, M. S. Uddin, M. Z. Parvez, and L. Nahar, "A squeeze and excitation ResNeXt-based deep learning model for Bangla handwritten compound character recognition," *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 6, pp. 3356–3364, 2022, doi: 10.1016/j.jksuci.2021.01.021.

[16] P. Mukherjee, S. Sen, K. Roy, and R. Sarkar, "Recognition of online handwritten Bangla characters using supervised and unsupervised learning approaches," *International Journal of Computer Vision and Image Processing (IJCVIP)*, vol. 10, no. 3, pp. 18–30, 2020, doi: 10.4018/ijcvip.2020070102.

[17] C. Saha, R. H. Faisal, and M. M. Rahman, "Bangla handwritten basic character recognition using deep convolutional neural network," *2019 Joint 8th International Conference on Informatics, Electronics and Vision (ICIEV) and 3rd International Conference on Imaging, Vision and Pattern Recognition (icIVPR)*, pp. 190–195, 2019, doi: 10.1109/ICIEV.2019.8858575.

[18] C. Saha, R. H. Faisal, and M. M. Rahman, "Bangla handwritten character recognition using local binary pattern and its variants," *2018 International Conference on Innovations in Science, Engineering and Technology (ICISET)*, pp. 236–241, 2018, doi: 10.1109/ICISET.2018.8745645.

[19] N. Sarika, N. Sirisala, and M. S. Velpuru, "CNN based optical character recognition and applications," *Proceedings of the 6th International Conference on Inventive Computation Technologies (ICICT)*, pp. 666–672, 2021, doi: 10.1109/ICICT50816.2021.9358735.

[20] K. K. Rabbi, A. Hossain, P. Dev, A. Sadman, D. Z. Karim, and A. A. Rasel, "KDANet: Handwritten character recognition for Bangla language using deep learning," *2022 25th International Conference on Computer and Information Technology (ICCIT)*, pp. 651–656, 2022, doi: 10.1109/ICCIT57492.2022.10054708.

[21] R. Pramanik and S. Bag, "Shape decomposition-based handwritten compound character recognition for Bangla OCR," *Journal of Visual Communication and Image Representation*, vol. 50, pp. 123–134, Jan. 2018, doi: 10.1016/j.jvcir.2017.11.016.

[22] N. Koiso, Y. Takemoto, Y. Ishikawa, and M. Takata, "Proposed method of acquiring train data for early-modern Japanese printed character recognizers," *Journal of Supercomputing*, vol. 81, no. 6, 2025, doi: 10.1007/s11227-024-06866-4.

[23] F. M. Rusli, K. A. Adhiguna, and H. Irawan, "Indonesian ID card extractor using optical character recognition and natural language post-processing," *2021 9th International Conference on Information and Communication Technology (ICoICT)*, pp. 621–626, 2021, doi: 10.1109/ICoICT52021.2021.9527510.

[24] H. Moussaoui, N. E. Akkad, and M. Benslimane, "License plate text recognition using deep learning, NLP, and image processing techniques," *Statistics, Optimization and Information Computing*, vol. 12, no. 3, pp. 685–696, 2024, doi: 10.19139/SOIC-2310-5070-1966.

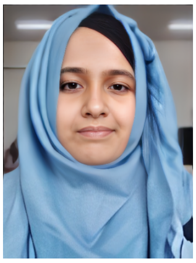
[25] S. Rajendran, M. A. Kumar, R. Rajalakshmi, V. Dhanalakshmi, P. Balasubramanian, and K. P. Soman, "Tamil NLP technologies: challenges, state of the art, trends and future scope," *Communications in Computer and Information Science*, pp. 73–98, 2023, doi: 10.1007/978-3-031-33231-9$_6$.

BIOGRAPHIES OF AUTHORS

Kazi Rifat Ahmed has completed his B.Sc. from Daffodil International University in Software Engineering and M.Sc. from the Institute of Information Technology, Jahangirnagar University. He is currently working as a lecturer in the Department of Software Engineering, Daffodil International University. His research interests are machine learning, deep learning, NLP, and computer vision. He has published in high-impact journals and conferences, aiming to advance AI-driven solutions in healthcare and security. He can be contacted at email: rifat.swe@diu.edu.bd.

Nusrat Jahan is working as an assistant professor and head at Department of Information Technology & Management in Daffodil International University, Bangladesh. She completed her M.Sc. and B.Sc. in Information Technology from Institute of Information Technology, Jahangirnagar University. She is doing her Ph.D. from Department of Computer Engineering, University Malaysia Perlis (UniMap). She is interested in technology management, computer networks, machine learning, and artificial intelligence. She can be contacted at email: nusrat.swe@diu.edu.bd.

Adiba Masud is currently pursuing her Ph.D. in the Department of Computer Science, University of Texas at San Antonio, Texas, USA. She has completed her B.Sc. and M.Sc. from the Institute of Information Technology, Jahangirnagar University. She is currently on study leave as a lecturer in the Department of Software Engineering, Daffodil International University. Her research interests are machine learning, deep learning, NLP, and computer vision. She can be contacted at email: adiba.swe@diu.edu.bd.



Nusrat Tasnim has completed her B.Sc. and M.Sc. from the Institute of Information Technology, Jahangirnagar University. She is currently working as a lecturer in the Department of Information and Communication Technology, Bangladesh University of Professionals. She was a former lecturer in the Department of Software Engineering, Daffodil International University. Her research interests are machine learning, deep learning, NLP, and computer vision. She can be contacted at email: nusrattasnim17@gmail.com.

Sazia Sharmin has completed her B.Sc. and M.Sc. from the Institute of Information Technology, Jahangirnagar University. She is currently working as a lecturer in the Department of Computer Science at the American International University, and she was previously a lecturer in the Department of Software Engineering at Daffodil International University. Her research interests are machine learning, deep learning, NLP, and computer vision. She can be contacted at email: sazia.sharmin@aiub.edu.

Nusrat Jahan Mim has completed her B.Sc. and M.Sc. from the Department of Software Engineering, Daffodil International University. She is currently working as a lecturer in the Department of Software Engineering, Daffodil International University. Her research interests are machine learning, deep learning, NLP, and computer vision. She can be contacted at email: nusratjahan.swe@diu.edu.bd.

Imran Mahmud received the master's degree in software engineering from the University of Hertfordshire, U.K., in 2008, and the Ph.D. degree in technology management from Universiti Sains Malaysia, in 2017. He is currently the head and an professor with the Department of Software Engineering, Daffodil International University, Bangladesh. He is also a visiting professor with the Graduate School of Business, Universiti Sains Malaysia. Previously, he was a senior lecturer with the Graduate School of Business, Universiti Sains Malaysia. He was a visiting lecturer with the Institute of Technology, Bandung, Indonesia, and the Hong Kong Management Association, Hong Kong. He achieved several awards, including the Hall of Fame and Prestigious Publication Award from Universiti Sains Malaysia, the young researcher from Kasetsart University, Thailand, and the young scientist in Technology Management from the Venus International Foundation, India. He can be contacted at email: imranmahmud@daffodilvarsity.edu.bd.