International Journal of Current Science Research and Review

ISSN: 2581-8341

Volume 05 Issue 02 February 2022

DOI: 10.47191/ijcsrr/V5-i2-17, Impact Factor: 5.825

IJCSRR @ 2022

www.ijcsrr.org

Real time Bangla Digit Recognition through Hand Gestures on Air Using Deep Learning and OpenCV

Chayti Saha[1], Fozilatunnesa Masuma[2], Khalil Ahammad[3], Chowdhury Shahriar Muzammel[4], Md. Mohibullah[5]

[1,2] Student, Department of CSE, Comilla University, Cumilla-3506

[3,4,5] Assistant Professor, Department of CSE, Comilla University, Cumilla-3506

ABSTRACT: Digit Recognition in real time through hand gestures has achieved great attention in machine learning and computer vision applications. This article focuses on identifying Bangla numerals in the air using hand motions. This research leads to the stairwell, allowing for more investigation in the same subject for various Bangla characters and even phrases. The major issue, however, is coping with the wide range of handwriting styles employed by various users. Many studies have been done on the identification of Bangla handwritten digits, but none has proven successful at recognizing Bangla digits in real time using hand gestures in the air. As a result, this article describes the creation of a Bangla digit recognition model that employs a Convolution Neural Network (CNN) to predict Bangla digits by observing hand movements in the air space. After a thorough examination, the suggested system attained a 98.37% accuracy on the BanglaLekha-Isolated dataset.

KEYWORDS: Convolution Neural Network (CNN); Human-computer interaction (HCI); Multilayer Perceptron (MLP); Rectified Layer Unit (ReLU); Real time Bangla Digit Recognition through Hand Gestures (RBDRHG).

INTRODUCTION

With approximately 200 million speakers [7,11], Bangla is a widely spoken language. It is Bangladesh's native and official language, as well as India's second most commonly spoken language. Bangla too has a long and distinguished history. International Mother Language Day was proclaimed by the United Nations Educational, Scientific, and Cultural Organization (UNESCO) on February 21st in memory of the language martyrs who perished in Bangladesh in 1952. Bangla characters are written in a Sanskrit-based system, which differs significantly from Latin or English scripts, making recognition jobs difficult to perform and reach the necessary accuracy. Following Table 1. show the interpretation of Bangla digit symbols in English digit symbols.

Table 1. The interpretation of Bangla digit symbols in English digit symbols

Bangla Digit Symbol	*Interpretation in English*
০ (শূন্য)	0 (Zero)
১ (এক)	1 (One)
২ (দুই)	2 (Two)
৩ (তিন)	3 (Three)
৪ (চার)	4 (Four)
৫ (পাঁচ)	5 (Five)
৬ (ছয়)	6 (Six)
৭ (সাত)	7 (Seven)
৮ (আট)	8 (Eight)
৯ (নয়)	9 (Nine)

Human-computer interaction requires the identification of hand gestures utilizing vision-based technology (HCI). A gesture is a visual depiction of a physical activity or an emotion. Between people and computers, gestures can be used to communicate. It differs from standard hardware-based methods in that it enables human-computer interaction through gesture recognition. Recognizing gestures or motions of the body or body parts, gesture recognition detects the user's intent. Hand motion detection technology has been the focus of many researchers for decades. Many applications rely significantly on hand gesture recognition, such as identification of sign languages, virtual and augmented reality, interpreters of sign language for the visually handicapped, and controlling a robot are just a few examples.

Directly or indirectly, efficient handwritten letter recognition systems are required for real-world tasks like controlling of touch-less screens, text digitalization, encounters with robots and more. As a result, it's important to recognize these characters which has been mostly studied throughout the previous decades of research. There are many research works related to this topic, even for Bangla language also, but none of them is done for real time Bangla digit recognition through hand gestures. Furthermore, for the reasons listed below, handwritten digit recognition in Bangla is more difficult than printed forms of character:

- Numbers written by distinct people are not only different, but they are also non-identical, range in size and form in a variety of ways.
- Individual numerals can be written in several different ways, making identification difficult.
- The challenge of recognizing numerals is further exacerbated by overlaps, which are identical digits in different forms.

In conclusion, due to the vast diversity of types of writing and the complication of characteristics of handwritten numbers, it is difficult to accurately categorize handwritten digits. In Bangla, there are ten digits. Furthermore, Bangla has a lot of numbers that are all the same form. This makes it difficult to increase performance using a simple classification method and impedes the creation of a dependable handwritten digit recognition system in Bangla. However, the lack of such real-time works in Bangla, as well as gestures, demonstrates the possibility to investigate this interesting field.

As a result, developing a method for detecting Bangla numbers is crucial [7-9]. In this paper, we suggested a real-time Bangla Digit identification system based on hand gestures. In order to meet these objectives, we developed the CNN model for predicting Bangla digits, as CNN is the largest English-language prediction model. The most well-known picture classification technique use cameras to identify hand movement in real time using OpenCV. There are relatively few open source datasets accessible online for Bangla Handwriting. The model was constructed using the BanglaLekha-Isolated dataset.

LITERATURE REVIEW

A few efforts have been done in the domain of real-time Bangla digit identification using hand gestures despite the fact that they were primarily for the Bangla sign language. Liu and Suen [3] showed directional gradient features for handwritten Bangla digit classification using the ISI Bangla numeral dataset [1], which comprises 19,392 training samples, 4000 test samples, and 10 classes (i.e., 0 to 9) Khalil Ahmed et al. [7] developed a model using the concept of CNN to recognize Bengali Sign Language gesture images for numerals in real time. Their suggested CNN model's maximum recognition accuracy with the dataset without the image rotation approach is 94.17%, while the recognition accuracy with the rotated images in the dataset is 99.75%. Surinta et al. [10] suggested a method with properties such as the form of the handwritten picture computed using 8-directional codes, the distance estimated between hotspots and black pixels, and the intensity of tiny block pixel space. The support vector machine (SVM) [2] classifier is fed each of the characteristics in turn, and the majority voting method is utilized to determine the final decision. Das et al. [9] used a genetic algorithm-based area sampling technique for local feature selection and obtained 97% accuracy on Handwritten Bangla Character Recognition. Among those who have contributed to this work are Xu and colleagues. [6] utilized a hierarchical Bayesian network to classify raw pictures using a bottom-up method. Sparse representation classifiers have also been used for handwritten digit recognition in Bangla [5], with 94% accuracy reported. Handwritten Bangla basic and compound letter recognition using Multilayer Perceptron (MLP) [11] and SVM classifier was proposed in [8] whereas handwritten Bangla number recognition using MLP was reported in [4], with an average recognition rate of 96.67%.

METHODOLOGY

This section starts with a description of the data gathered and then moves on to the full process of the experiment. The data was pre-processed as needed, such as picture resizing, grey scaling, and augmentation. The working method of the suggested CNN model was presented.

Fig. 1 depicts the entire process, whereas Fig. 2 depicts the model block diagram. Fig. 1 depicts a high-level perspective of the proposed system. There are two stages to the model's design: The first step is for pre-processing digit pictures captured by a camera that records hand gestures in real time, and the second level is for classification using the CNN model. To make progress, the classification precision of the proposed CNN model, we included a pre-processing module that includes segmentation, morphological dilation and erosion, Gaussian smoothing filter, clipping, and normalization. To recognize the numbers in real time, the model was saved and imported into OpenCV.

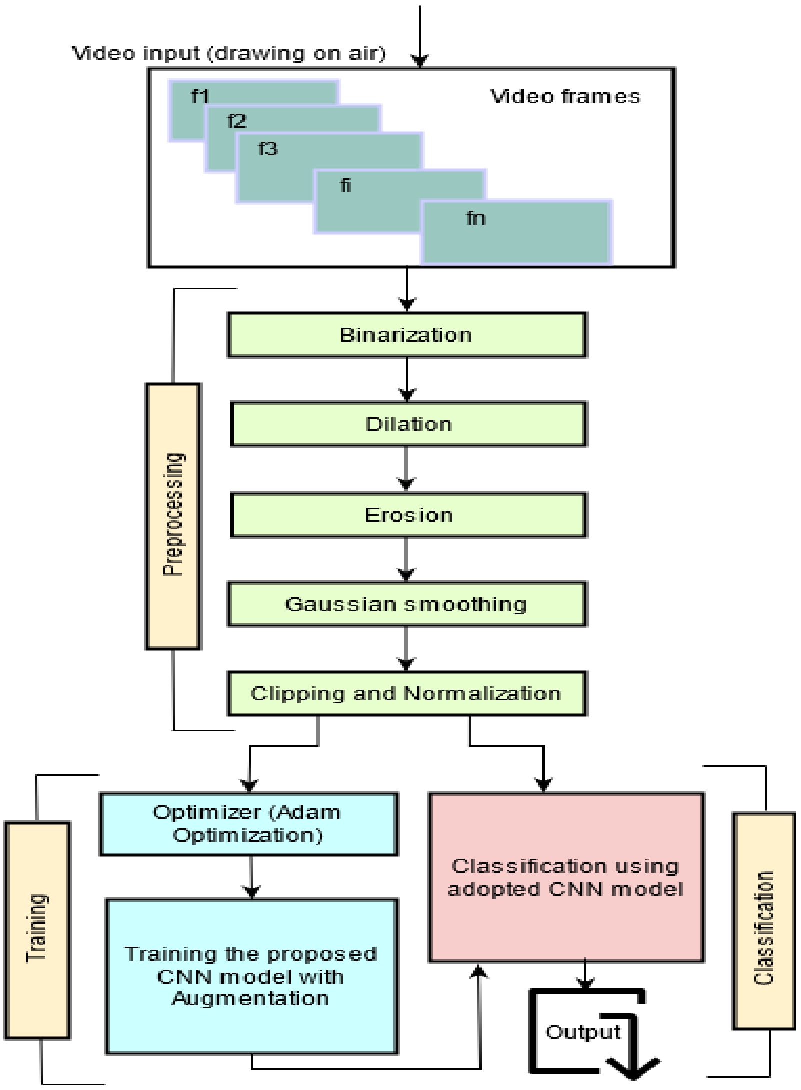

Video input (drawing on air)
Video frames
f1
f2
f3
fi
fn
Preprocessing
Binarization
Dilation
Erosion
Gaussian smoothing
Clipping and Normalization
Training
Optimizer (Adam Optimization)
Training the proposed CNN model with Augmentation
Classification using adopted CNN model
Output
Classification

Fig. 1. Block diagram of proposed system

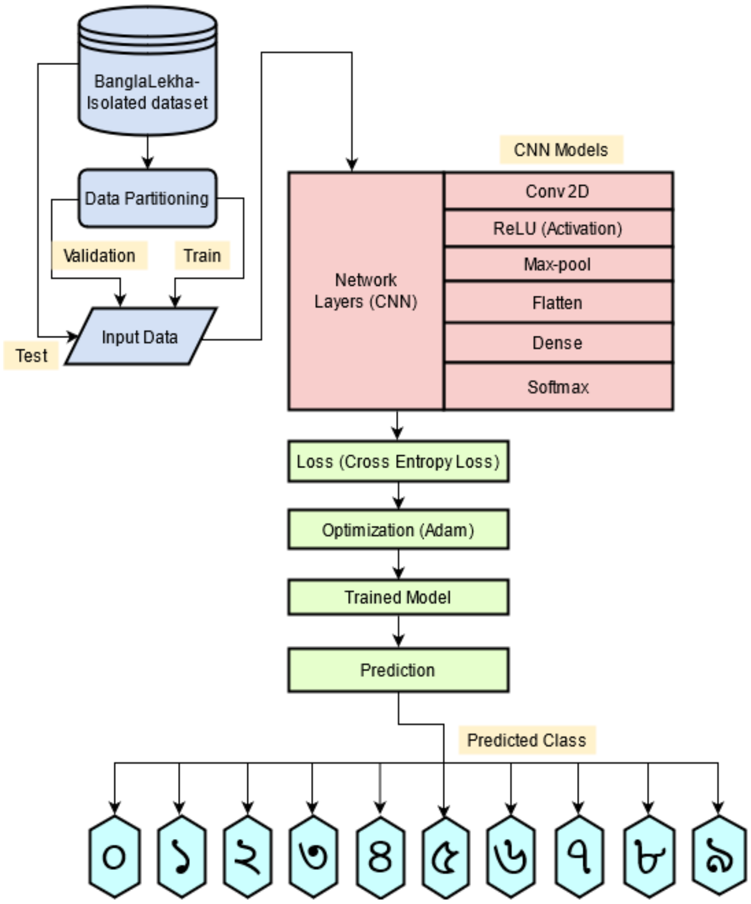


Fig. 2. Flow diagram of model implementing Bangla hand written digit recognition

A brief overview of each module is discussed below:

DATA COLLECTION

We used the available dataset BanglaLekha-Isolated to perform our job. It is a solitary dataset with handwritten characters in Bangla. This BanglaLekha-Isolated dataset includes 84 unique characters, including 50 fundamental characters, 10 numbers, and 24 compound characters in Bangla. For each of the 84 characters, 2000 handwriting samples were gathered. After removing errors, the collection was reduced to 1,66,105 handwritten character images. This dataset might be used to investigate how gender and age affect penmanship as well as optical handwriting recognition studies. Only pictures of numbers from the whole dataset were utilized in our system. The picture is inverted, scaled, and padding is added to make it square while keeping the aspect ratio intact. A total of 13,748 photos are used to create the train set, while 4000 and 2000 images are used to create the test and validation sets, respectively.

DATA PREPARATION

In real-time, the suggested method extracts isolated Bangla handwritten digits' pictures from a camera. The obtained pictures are then converted to binary images. The picture is next processed with morphological dilation and erosion processes, followed by a Gaussian smoothing filter to make it noise-free and robust. Each frame is given to the previously trained CNN model for classification after scaling (to 28*28 pixels) and padding for digit identification.

We first prepared the dataset in order to train the model. We started by correcting some of the dataset's inaccurate labeling pictures, then deleted some of the wrong photographs, such as blank images. Because we utilized the BanglaLekha-Isolated dataset, which already included inverted pictures, it came with edge thickening, reversed foreground and background, median filter noise reduction.

IMAGE RESIZING AND GREY SCALING

Prior to starting modeling, the pictures will need to be modified so that they all have the same form. Each picture in the BanglaLekha-Isolated dataset was a distinct size. As a result, we reduced the image size to 28*28 pixels. This RGB image has to be converted to grayscale in order to create a binary image. However, inverted pictures are already included in the collection. The image may lose some information as a result of resizing. After resizing the pictures, we utilized the Lanczos interpolation technique.

IMAGE AUGMENTATION

Using the Data Augmentation method, the performance of the suggested model is enhanced. In order to increase the model's performance, we used Data Augmentation methods. For Data Augmentation on training pictures, the following techniques were chosen:

- Randomly shift the height and the width of the images 20%.
- Randomly rotate the training images 20°.
- Randomly zoom and shear the training images 20%

To avoid overfitting and broaden the dataset, we utilized Data Augmentation. These modifications allowed us to add some variety to the dataset, which can happen if the digits are written by someone else. Fig. 3 depicts the original, rotated, shifted, zoomed and sheared image of digit "২".

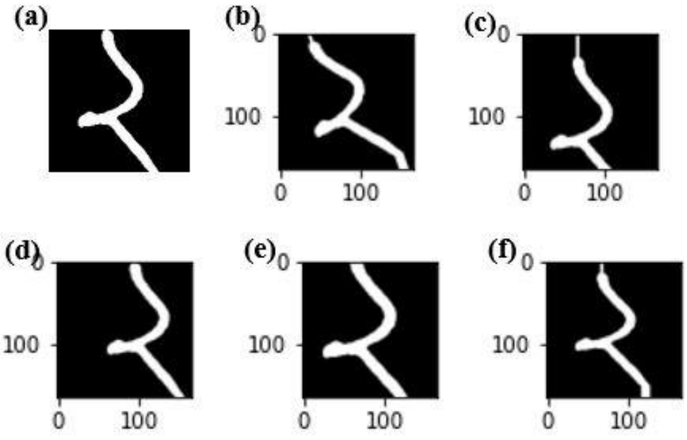


Fig. 3. (a) Original Image (b)Rotated Image (c) Height shifted image(d)Width shifted image (e)Zoomed image (f) sheared image

MODEL DEVELOPMENT

Recognition of patterns and image analysis issues frequently employ neural networks. CNN, a deep learning approach, the neuron connection pattern as an inspiration in visual cortex of animals, is the most promising instrument for doing so. It's mostly used for image identification, object recognition, and other similar tasks. The KERAS Python library and TensorFlow as a backend are used to build the CNN model for identifying numbers. As a classifier, a Sequential Model is employed, which is made out of a layer stack that runs in a straight line. Neurons in CNN contain biases and weights that can be learned. CNN requires the least amount of preprocessing when compared to other algorithms. The input to a neural network is always a vector, while with a CNN, the input is a multi-channeled image. An input layer, concealed layers, and an output layer make up a CNN. The hidden layer consists of the convolutional layer, Rectified layer unit (ReLU), which is the activation function, pooling layers, normalized layers, and completely linked layers.

International Journal of Current Science Research and Review
ISSN: 2581-8341
Volume 05 Issue 02 February 2022
DOI: 10.47191/ijcsrr/V5-i2-17, Impact Factor: 5.825
IJCSRR @ 2022
www.ijcsrr.org

The neurons in CNN have biases and weights that can be learned. Preprocessing time for CNN is the shortest compared to other methods. In a neural network, the input is always a vector, but in a CNN, the input is a multi-channeled picture. An input layer, concealed layers, and an output layer make up the CNN.

Figure 4 depicts a CNN's general design, which consists of two primary components: a feature extractor and a classifier. Each layer (input layer) in the feature extraction unit the output is sent to the intermediate layer below it, which is regarded as the preceding layer's output layer, and the immediate next layer receives the current output as input, whereas, based on the input data, the categorization section provides anticipated outcomes. The essential layers of CNN architecture are the convolution and pooling layers. Each node in the convolution layer uses a convolution operation done to the input nodes to extract features from the input pictures.

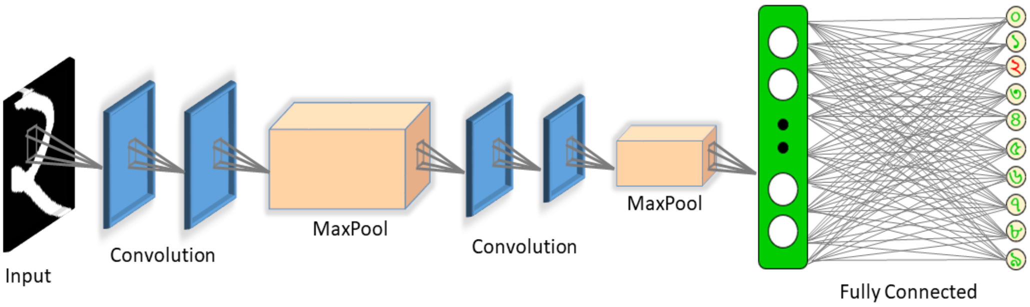


Fig. 4. CNN for digit recognition

Through average or maximum operations performed to input nodes, the max-pooling layer serves to tabulate the feature. The outputs of the (l - 1)th layer are employed as input for the lth layer. The inputs pass through a series of kernels before being processed by a nonlinear function known as ReLU. Suppose, f refers to activation function of ReLU. As an example, if x_i^{l-1} inputs from (l - 1)th layer, $k^l_{i,j}$ are kernels of lth layer. The favoritism of lth layer are illustrated with b^l_j. Then, convolution operation can be expressed as following:

$$x_j^l = f\left(x_i^{l-1} * k_{i_1 j}^l\right) + b_j^l \qquad (1)$$

The subsampling or pooling layer abstracts this feature, which performs average or maximum operations on the input nodes. For instance, each output dimension of a 2∗2 down sampling kernel with half of the size of their respective input dimension. The following is an example of a pooling operation:

$$x_j^l \ = down\left(x_i^{l-1}\right) \qquad (2)$$

Unlike conventional Neural Networks, CNN retrieves low to high level characteristics (NN). The higher level characteristics are originated in the propagating features of the lower level layers. As features spread to the topmost layer, the feature's dimensions can be decreased based on the dimensions of the convolution and pooling masks. However, to improve classification accuracy, the number of features mapped is generally increased to choose or map the most appropriate characteristics of the input pictures. The fully connected network receives its inputs from the outputs of the final layer of CNN. It employs a Softmax technique to provide categorization outputs. For any input sample x, any weight vector w, and distinct linear functions k, the Softmax operation can be defined for the ith class as follows:

$$P(y = i|x) = \frac{\exp x^T\,\omega_{\mathrm{i}}}{\sum_{k=1}^{K} \exp x^T\,\omega_{\mathrm{k}}} \qquad \mathbf{(3)}$$

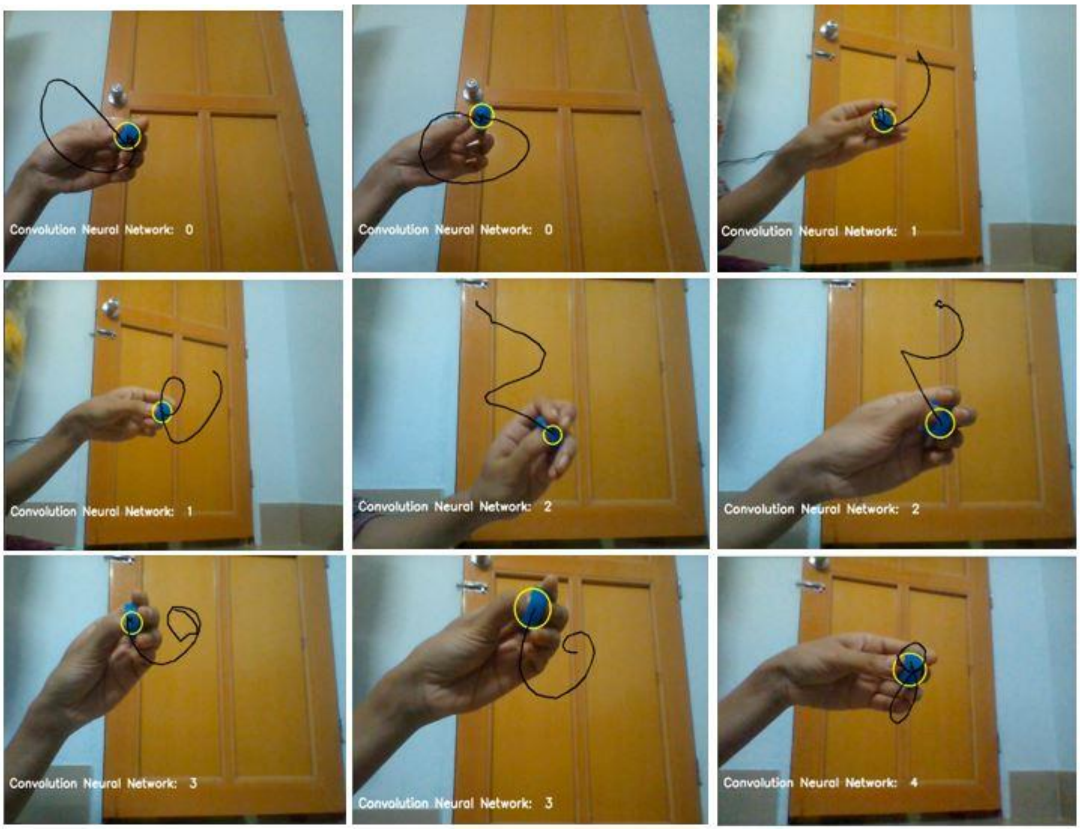
The suggested architecture comprises two convolutional layers and two dense layers that are completely linked. There were 32 filters in the first convolution layer and 64 filters in the second layer. All layers were activated using the Rectified Linear Unit (ReLu) (V.Nair and G. E. Hinton, 2010). There were two Max Pooling layers to choose from. The max-pooling layer has a pool size of 22. The first of the two completely linked layers contained 128 filters, while the second had 10 filters for the 10 digits. For the categorization, the last activation function was a Softmax function. We updated the weights using Adam's optimizer. The model overview of the CNN architecture that has been proposed is shown in Table 2.

Table 2. The model summary of the proposed CNN architecture

Layer (type)	*Output Shape*	*Param*
Conv2D	(None, 26, 26, 32)	320
MaxPooling2D	(None, 13, 13, 32)	0
Conv2D	(None, 11, 11, 64)	18496
MaxPooling2D	(None, 5, 5, 64)	0
Flatten	(None, 1600)	0
Dense	(None, 128)	204928
Dense	(None, 10)	1290

IMAGE RECOGNITION

Using TensorFlow as a backend, we loaded the stored trained model. We used Python as the editor and programming language, and OpenCV to read the video frame by frame from a camera in real-time (using a while loop). With the help of a blue colored item, the user may write Bangla numerals on the screen. We seek for a blue-colored item inside the frames as soon as the camera starts to read. Once we've located the contour (blue object), we smooth it out using a number of image operations before drawing on the screen with the contour's center. Finally, the center is saved as a deque so that the system may put them all together to make a whole writing. When the user has finished writing, the system combines the points it had previously saved, displays them on a blackboard, rescales them to 28*28*1 pixels, and sends them to the models. The model detects and forecasts the Bangla digits correctly. As illustrated in Fig. 5, our suggested system converts Bangla numerals to English digits and displays them on a screen.


Convolution Neural Network: 0
Convolution Neural Network: 0
Convolution Neural Network: 1
Convolution Neural Network: 1
Convolution Neural Network: 2
Convolution Neural Network: 2
Convolution Neural Network: 3
Convolution Neural Network: 3
Convolution Neural Network: 4

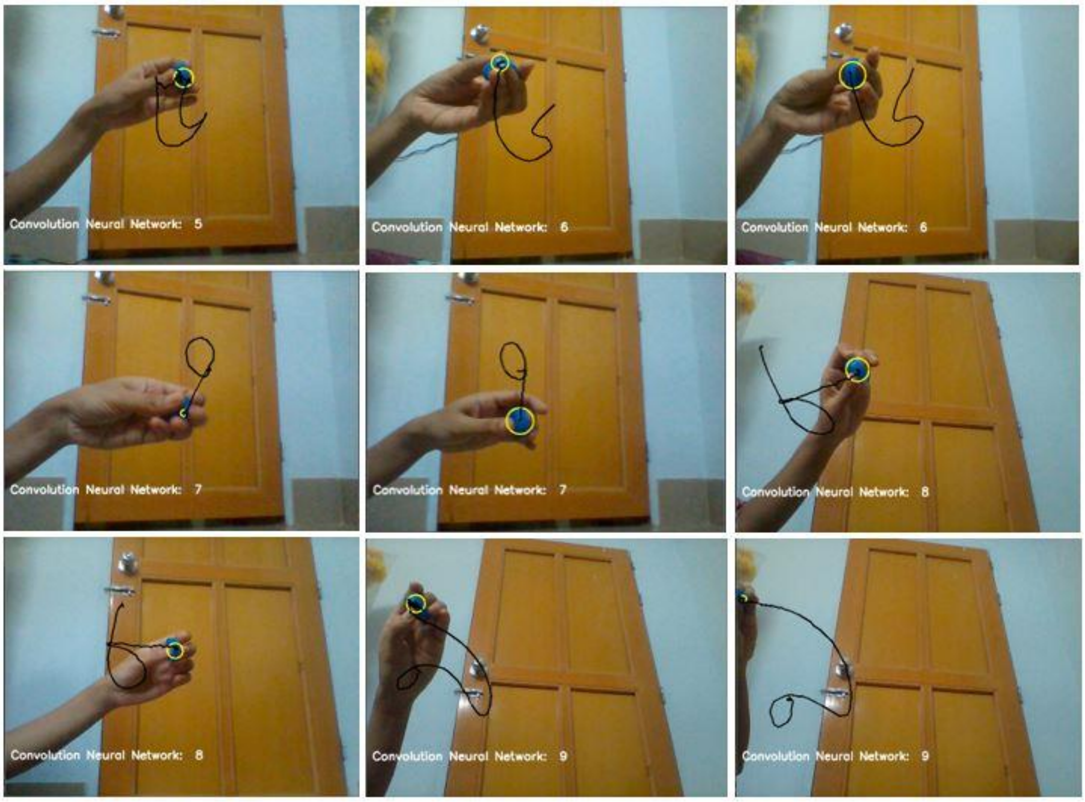

Convolution Neural Network: 5
Convolution Neural Network: 6
Convolution Neural Network: 6
Convolution Neural Network: 7
Convolution Neural Network: 7
Convolution Neural Network: 8
Convolution Neural Network: 8
Convolution Neural Network: 9
Convolution Neural Network: 9

Fig. 5. Digits (09) output

EXPERIMENTAL RESULT ANALYSIS AND PERFORMANCE EVALUATION

In the experiment, the dataset was split into training, validation, and testing using a split ratio of roughly 70%, 10%,20%. Our experiment was finished with a training accuracy of 93.29%, validation accuracy of 96% and test accuracy of 98.37%. Fig. 6 visually represents the train data, test data, validation data ratio.

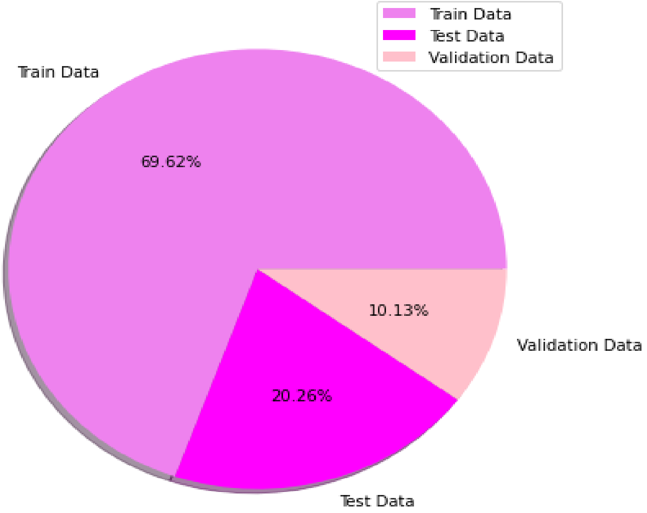


Fig. 6. Dataset splitting ratio

We have used 34 epochs for training the designed model. After completing 34 epochs, the training came to an end. Training accuracy is superior than validation accuracy and test accuracy. Because our approach can detect rotated, shifted, zoomed, superimposition, and occluded images since our model was taught to recognize them. Without using augmentation, we achieved 99.36% training accuracy, 97% validation accuracy and 97.30% test accuracy without implementing augmentation. The model could not recognize enhanced real time inputs despite having higher training accuracy than validation and test accuracy.

We acquire our results after the system was unable to identify and validate after training. The validation dataset, which is a data pattern generated by our trained model, was used. We utilize it to estimate model skill while tweaking hyper parameters. Table 3. shows the training loss, training accuracy, validation loss, validation accuracy for some epochs, whereas we have used 34 epochs. Figure 6 depicts the accuracy of our model's training and validation, whereas Fig. 7 depicts the accuracy of our model's training and validation, whereas Fig. 8 depicts the loss of training and validation, as shown below.

Table 3. Train,Validation loss and Train,Validation Accuracy for epochs

Epoch	*Training Loss*	*Training Accuracy*	*Validation Loss*	*Validation Accuracy*
1	1.0733	0.6407	0.6124	0.7913
5	0.4495	0.8510	0.2514	0.9163
10	0.3234	0.8932	0.2019	0.9320
15	0.2939	0.9046	0.1766	0.9380
20	0.2473	0.9181	0.1686	0.9441
25	0.2360	0.9213	0.1418	0.9531
30	0.2182	0.9268	0.1134	0.9602
34	0.2107	0.9312	0.1344	0.9541

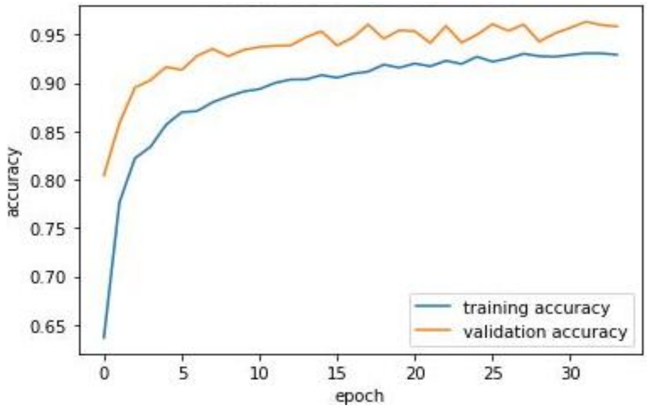


Fig. 7. Training and validation accuracy

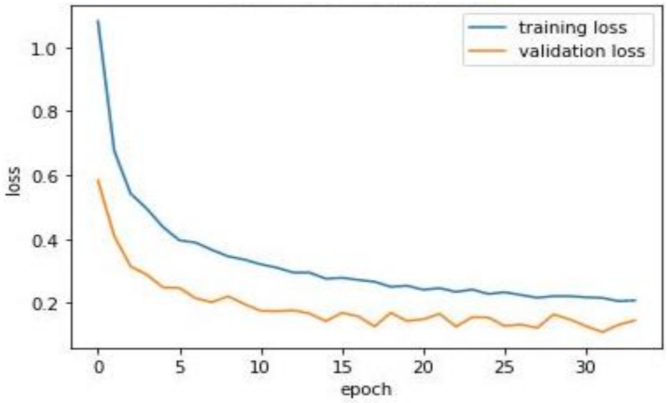


Fig. 8. Training and validation loss

The blue and yellow curves in the above figures represent training accuracy and training loss, respectively, while the red and yellow curves represent validation accuracy and validation loss, respectively.

The performance of the developed CNN model is shown in Fig. 9 by the confusion matrix. The most frequently misclassified digit was 1, which was misclassified 14 times as number 9. The numerals 1 and 9 were shown to have the largest false negative and false positive, respectively. With ten classes of digits, digit 9 was predicted wrongly 16 times, and digit 1 was predicted incorrectly 9 times. The misclassification of the digits was caused by similarities in the digits' forms and varied angles when the images were captured.

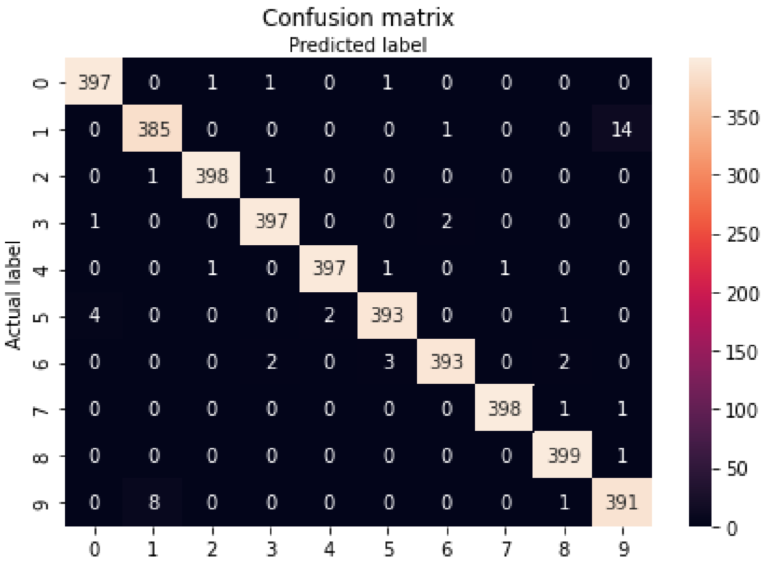


Fig. 9. Confusion Matrix

The precision, recall, and F1-score for each class are shown in Table 4. Class 9 has the lowest precision of 0.96, the lowest recall of 0.96, and the lowest F1-score of 0.97 for both 1 and 9. The weighted average, on the other hand, was boosted to 99%.

Table 4: Precision, Recall, F1-score for each digit

Class	Precision	Recall	F1-score
0	0.99	0.99	0.99
1	0.98	0.96	0.97
2	0.99	0.99	0.99
3	0.99	0.99	0.99
4	0.99	0.99	0.99
5	0.99	0.98	0.98
6	0.99	0.98	0.99
7	1.00	0.99	1.00
8	0.99	1.00	0.99
9	0.96	0.98	0.97
Weighted Avg	0.99	0.99	0.99

CONCLUSION

Bangladesh’s official language is Bangla, spoken by 300 million people worldwide, making it the world's seventh most fluently spoken native language. Despite improvements in object recognition technology, the problem of Real-time Bangla Digit Recognition via Hand Gestures (RBDRG) remains unresolved owing to several complications. In actuality, even many advanced existing approaches do not produce satisfying results.

This study proposes a CNN architecture for handwritten digit identification written on air collected using a camera, which performs well in identifying most of the input digits in real time. The results indicate a test accuracy rate of 98.37% for identifying 10 Bangla digits on an integrated computer with limited computing capabilities and minimal power consumption. This definitely provides a favorable proof on the part of the simple CNNs that they are capable of solving rather complicated classification issues. Furthermore, this study effort serves as a reference for future initiatives that will shed light on this subject for future researchers working with machine learning approaches. In the future, we plan to develop our study towards Real-Time Bangla.

REFERENCES

1. B. Chaudhuri, “A complete handwritten numeral database of Bangla–amajor Indic script,” in Proceedings of Tenth International Workshop onFrontiers in Handwriting Recognition, Suvisoft, Baule, France, October2006.
2. C. Cortes and V. Vapnik, “Supportvector networks,” Machine Learning,vol. 20, no. 3, pp. 273–297, 1995.
3. C.L. Liu and C. Y. Suen, “A new benchmark on the recognition ofhandwritten Bangla and Farsi numeral characters,” Pattern Recognition,vol. 42, no. 12, pp. 3287–3295, 2009J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
4. D. E. Rumelhart, J. L. McClelland, P. R. Group et al., Parallel DistributedProcessing, vol. 1, MIT Press, Cambridge, MA, USA, 1987.
5. H. A. Khan, A. Al Helal, and K. I. Ahmed, “Handwritten Bangla digitrecognition using sparse representation classifier,” in Proceedings of2014 International Conference on Informatics, Electronics and Vision(ICIEV), pp. 1–6, IEEE, Dhaka, Bangladesh, May 2014.
6. J.W. Xu, J. Xu, and Y. Lu, “Handwritten Bangla digit recognition usinghierarchical Bayesian network,” in Proceedings of 3rd InternationalConference on Intelligent System and Knowledge Engineering, vol. 1,pp. 1096–1099, IEEE, Xiamen, China, November 2008.
7. Khalil Ahammad, Jubayer Ahmed Bhuiyan Shawon, Partha Chakraborty,Md Jahidul Islam, Saiful Islam, “Recognizing Bengali Sign LanguageGestures for Digits in Real Time using Convolutional Neural Network,”International Journal of Computer Science and Information Security,Vol. 19 No. 1 JANUARY 2021.
8. N. Das, B. Das, R. Sarkar, S. Basu, M. Kundu, and M. Nasipuri,“Handwritten Bangla basic and compound character recognition usingMLP and SVM classifier,” 2010.
9. N. Das, R. Sarkar, S. Basu, M. Kundu, M. Nasipuri, and D. K. Basu, “Agenetic algorithm based region sampling for selection of local featuresin handwritten digit recognition application,” Applied Soft Computing,vol. 12, no. 5, pp. 1592–1606, 2012.
10. O. Surinta, L. Schomaker, and M. Wiering, “A comparison of featureand pixelbased methods for recognizing handwritten Bangla digits,” inProceedings of 12th International Conference on Document Analysis andRecognition (ICDAR), pp. 165–169, IEEE, Buffalo, NY, USA, 2013.
11. S. Basu, N. Das, R. Sarkar, M. Kundu, M. Nasipuri, and D. K. Basu, “AnMLP based approach for recognition of handwritten Bangla numerals,”2012.
12. U. Pal and B. Chaudhuri, “Automatic recognition of unconstrainedoffline Bangla handwritten numerals,” in Proceedings of Advancesin Multimodal Interfaces–ICMI 2000, pp. 371–378, Springer, Beijing,China, October 2000.

***Cite this Article**: Chayti Saha, Fozilatunnesa Masuma, Khalil Ahammad, Chowdhury Shahriar Muzammel, Md. Mohibullah (2022). Real time Bangla Digit Recognition through Hand Gestures on Air Using Deep Learning and OpenCV. International Journal of Current Science Research and Review, 5(2), 435-445*