HANDWRITTEN DIGIT IDENTIFICATION USING CONVOLUTIONAL NEURAL NETWORK

Mainak dey[1], Sayandip Kumar[2], Anurupa Paul[3], Kaushiki Mitra[4], Pranab Hazra[5]

[5]*Assistant Professor in the Department of Electronics & Communication Engineering, Narula Institute of Technology, Agarpara, Kolkata, West Bengal, India.*

[1, 2,3,4]*Student of B-TECH in the Department of Electronics & Communication Engineering (ECE), Narula Institute of Technology, Agarpara, Kolkata, West Bengal, India.*

Abstract: *Here in this paper, we have presented a handwritten digit's identification system with the help of Convolutional Neural Network which a basic deep learning technique. Handwritten digit identification system solves pattern application related issues like postal mail sorting, bank cheque processing and data entry form etc. Here we have experimented deep learning method CNN to our MNIST test dataset and created a model which can make the user written digits recognizable to our machine. We are using WIN32GUI to draw a digit and classify it through CNN. So, the main target of our work is to implement a model to identify the handwritten digit. The model is trained using MNIST dataset and Convolutional Neural Network.*

Keywords: **Handwritten digit recognition, CNN, MNIST, Win32GUI, Keras module.**

1.INTRODUCTION

Images are easily recognized and being analyzed by the human brains. When we see a particular image, the brain divides it into different part and then analyses it. Image processing is a branch of computer science that attempts to do the identical thing with computers. Image processing converts images into digital form such as matrices which is machine readable and displays the result in categorize wise algorithm. Optical Character Identification (OCI) is a sub-field of processing of image it helps to recognize character from image and its primary application is to identify handwritten images [18]. These networks have the great ability of processing recognizing and learning images. CNN architecture for image categorization is classified into two different layer one is convolution layer for extracting images and another is connected layer for and connected layer it works on the classified feature extracted by the convoluted layer [4]. Handwritten digit identification is a very popular and interesting topic in recent years. Except this there are many other convolution network algorithms for handwritten digit identification in accuracy computation time sill require much more improvement. In literature there are various processes that have been used for handwritten digit identification.

2.RELATED WORK

The work of Handwriting digit Identification has an active community of academics studying it. A lot of important work on CNN happened for handwritten digit identification [2]. Now a days detection of handwritten Digits is playing a great role on various smart activities based on Machine Learning (ML). Some of this word includes Online Identification of Photographs containing Numerical Digits, Offline recognition, Real-Time Handwriting Recognition, Signature Verification, Postal-Address Interpretation, Bank-Check Processing, Writer Recognition. These activities are already used in Google Scanner for verification of numbers or digits.

3.DATA PREPARATION

For considering the initial step in Machine Learning classification models data Preparation is one of the most important steps. In some cases, the noise, and other types of errors in the dataset should be corrected before applying CNN [1].

Clearing and formatting are needed for data that contains unnecessary details. For ensuring that the data was free from null, and other unnecessary data, and before implementing CNN to classify (MNIST) handwritten digits, first we should complete the data preparation as an essential of the primary step to guarantee this dataset was fit to use in CNN [11]-[12]. Hereby we can consider MNIST as the primary step for implementation of any kind of ML programming. If we don't apply pre-processing to the raw data, there arise a high chance of possibility of unnecessary data leads to misleading results, except this there will also be negative affect in the classification accuracy.

3.1 Feature Density

In MNIST, it is consisting of a grid of 28 rows and 28 columns. So, this training and testing set consists of 784 numerical features. The names of this features are started from pixel 0 to pixel 783.For easier working we use the simplest form of this train and test set, for this we consider a distribution of 16 features only.

3.2 Mean and Standard Deflection of Train and Test Sets

From the below figure, we can notice that the values of testing and training sets are quite closer, which specify that the means of the features in and testing and training data are very close. In this figure, most of the value's precipitate on the central peak, which determines the values in testing and training sets are equally distributed. From the next figure, we can assume that the training and testing data distribution tends to be close to the mean, which determines the standard deviation of the given data is close to zero.

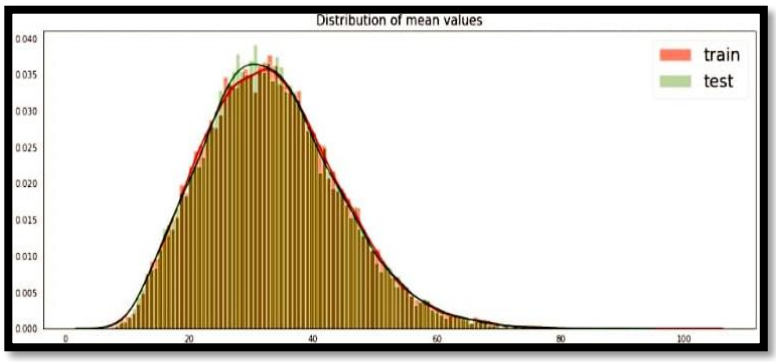


Figure 1. Distribution of the mean values of the testing and training sets [3]

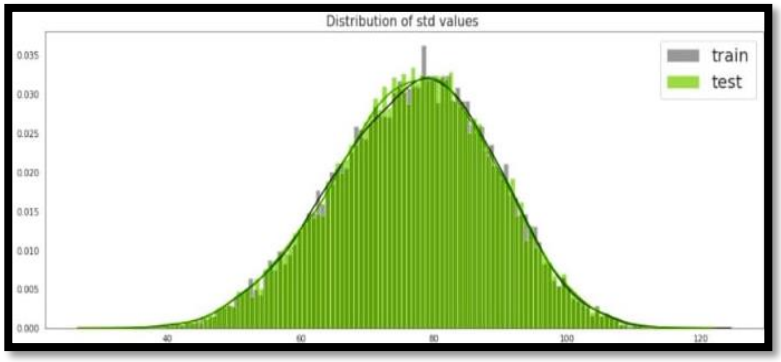


Figure 2. Distribution of the standard deviation of the and testing and training sets [5]

3.2 Mean Values of Labels in Training Set

From Figure 3, we can observe that the ten different labels (0 to 9) have their different averages, and this is quite logical because all labels have their different features. As expressed in Figure 3, the different averages remain unchanged with the distribution of these labels.

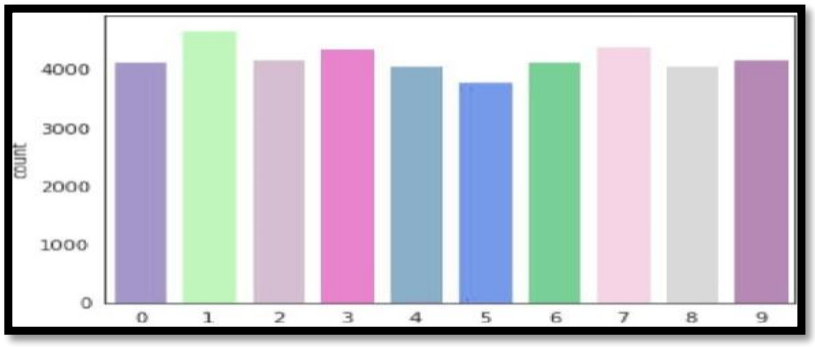


Figure 3. Distribution of mean values in the train set [7]

4. MATERIALS

Here the (MNIST) Modified National Institute of Standards and Technology dataset is treated as one of the most common and useful datasets, which consists images of handwritten digits in binary language [15]. MNIST dataset made up of 70,000 grayscale images of size 28 × 28. So, there are totally 784 features in the testing and training set.

4.1 CONVOLUTIONAL NEURAL NETWORK (CNN)

Convolution Neural Network is also known as CNN or a component of an artificial neural network that is so far been most popularly used for analyzing images. Although the image analysis has been the most widespread of use in CNN, they can also be utilized for other data analysis or classification problems as well. Most generally we think of a CNN as an artificial neural network (ANN) that has been some type of specialization for being able to pick out or identify patterns and make some sense of them. This pattern detection makes CNN so much useful for Image analysis. So, CNN is just some form of artificial neural network which differentiate it from a standard multilayer perceptron or MLP. CNN has hidden layers and these layers are accurately what makes a CNN. The main principle of CNN layer that is it receives input then transforms the input in some way and then the output transforms input in the next layer of convolutional layer [10]. Convolutional networks are able to take patterns and images more precisely, but the convolutional layers should be specified. Also in our project, for the handwritten digits CNN classify the digits into their respective categories of whether the images of 1,2,3 etc.

The most advantage of using CNN is that it shortens the number of parameters and speed up the training of model significantly**.**

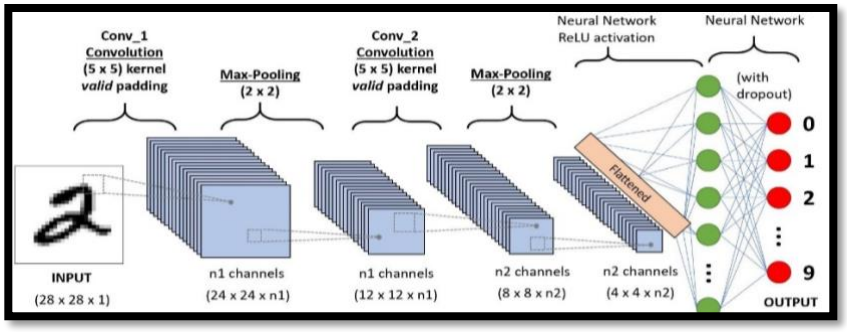


Figure 4. Working of CNN [19]

4.2 WIN32GUI

Win32 GUI is a platform where the native graphical user terminal toolset for Perl. Basically, it's having an XS application of most of the functions established in user32.dll and gdi32.dll, with an object-oriented Perl interface and an event-based dialog model that mimic the capability of visual basic. Win32 is the main set of Microsoft Windows API which is utilized for developing the 32-bit applications.

On the other hand, Graphical User Interface is a point where two systems do interact, and SPOC is a general example of interface. UI is an interface created for user of machine but there must be user interacting helps users to interact with devices like computers with similar interface such as screen [8]. Graphical representative elements on any website are ideal example of GUI [6]. GUI shows us what functions are possible. It verifies the objects showed on the screen and testing the visual elements of the application testing supports for delivering high quality and user-friendly software. In the end we can achieve a higher level of user engagement and satisfaction as it ensures the images and other features are user friendly or not. We can check clarity and pixel size. The most convenient part of using GUI is that we can catch offending elements in our UI, the quicker they can be resolved it efficiently reduces the number of risks towards the development of life cycle.

5.PROPOSED METHODOLOGY

- Step 1: Prepare the Handwritten Data Set.
- Step 2: Preprocess the data.
- Step 3: Import the library files.
- Step 4: Load the data set for detection.
- Step 5: Divide the data set into two-part training dataset and test data set.
- Step 6: Use the specific model based on deep learning for training purpose using this train data set.
- Step 7: Check the model accuracy by testing data set
- Step 8: Show the output for a specific input.

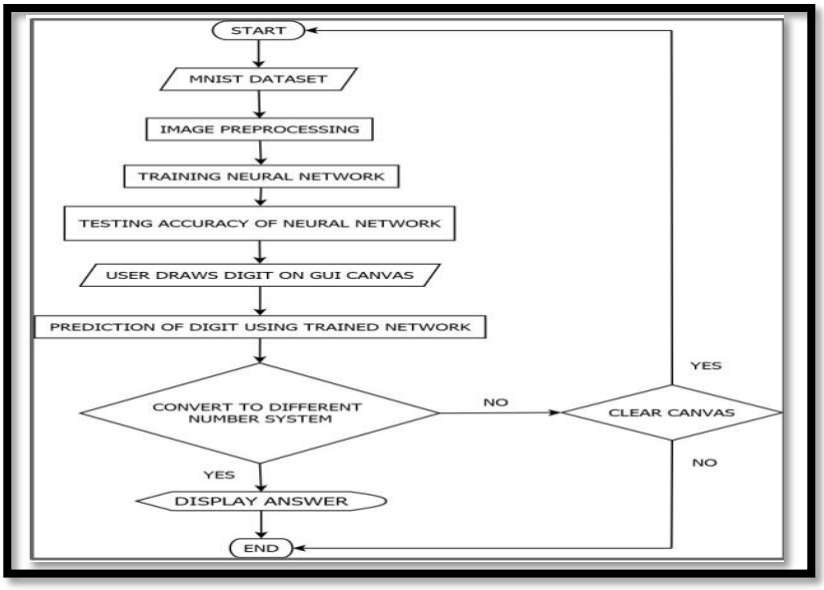


Figure 5. System Framework

6.EXPERIMENT METHODOLOGY

6.1 Experimental Data

By using this testing data, we can start evaluation our Model. Some examples of classification output from our model during testing came out. Some of this data values are mentioned below. Here we can also find out different test cases where the classifications are failed. But if we increase the training set and increased developed standard number of pattern then most pattern can be solved out in an easier way and more efficiently [14]. Misclassifications are also occurred due to the problem of image compression and image sharpness caused due to the missing pixels.

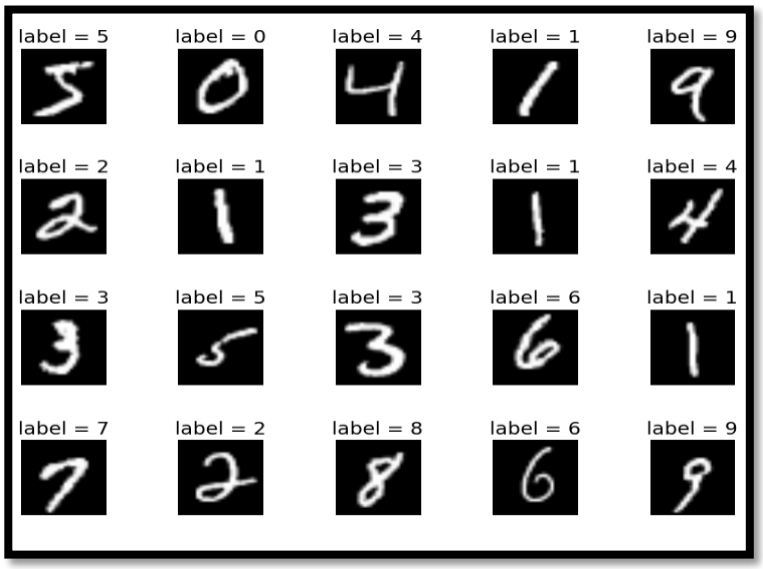


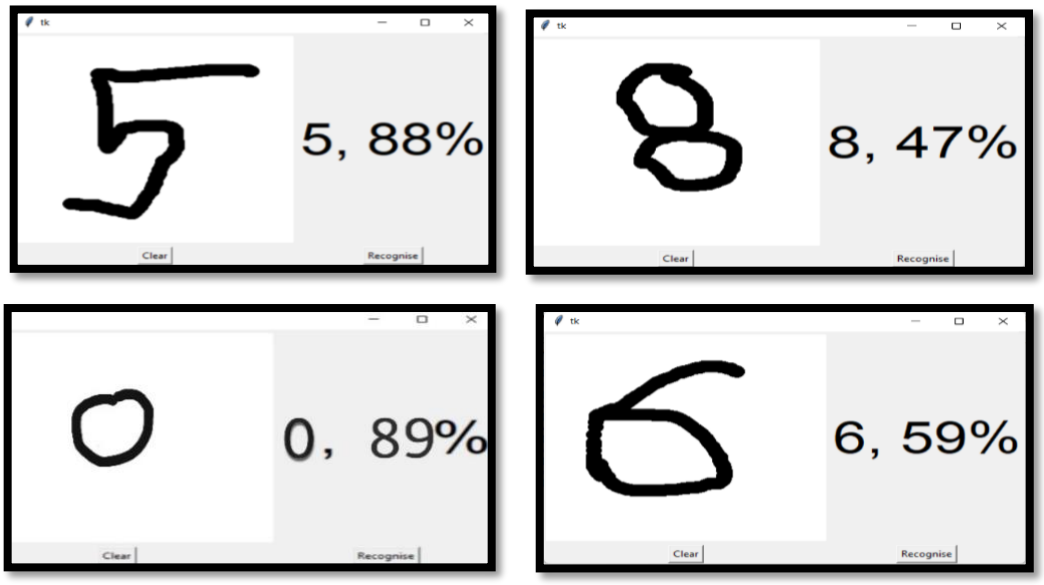
Figure 6. Digit Data input

6.2 Model Evaluation

At first, we will train the trainer kit based to the GUI. First, we should include Keras module which is a basic python library function for Deep Learning. Then we must import databases of MNIST with the help of Keras. Then other modules like Sequential, Dense, Dropout, conv2D etc. Then we must split the data between trainer set and test set. Then we must reshape the trainer set and test set according to (28 x 28) grid of MINST. Then class vectors are modified into Binary class Matrices. After modulations of all the library functions must compile accordingly [16]. Thus, our Model is trained successfully, and we will save this model as an ".h5" document.
For the output screen we must configure the GUI model. Here Tkinter, pypiwin32, NumPy must be imported. Firstly, we must load the MNIST from the ".h5" document which is created earlier. Then we have to call a function to predict the image. Then we will resize it to 28 x 28 pixel. As grayscale image is only accepted from MNIST we have to change the RGB color code to grayscale. After reshaping to support our model input and normalizing it we must predict the class [9]. Within the class we must define the X and Y index for implementing the picture. Then we have to create the element where the background color, text color, Hight and width along with the cursor type of the output screen is to be determined. Lastly, we must classify the handwriting and displaying the result with the help of Pipywin32.

7.ANALYSIS OF RESULTS

Among several test cases our model has conveniently identified many images with a high accuracy [17]. These reports are quite better for search a simple model just made u with CPU trainer and consumes a very less time for training (less than 20 minute) [13]. Our model can recognize all digits efficiently with good accuracy.
If the image of the digit which are inputted for detection are written in a bad handwriting, then also our model is capable to classify the figure with a good accuracy (82.5%). The result of this detection is mentioned next in the project.



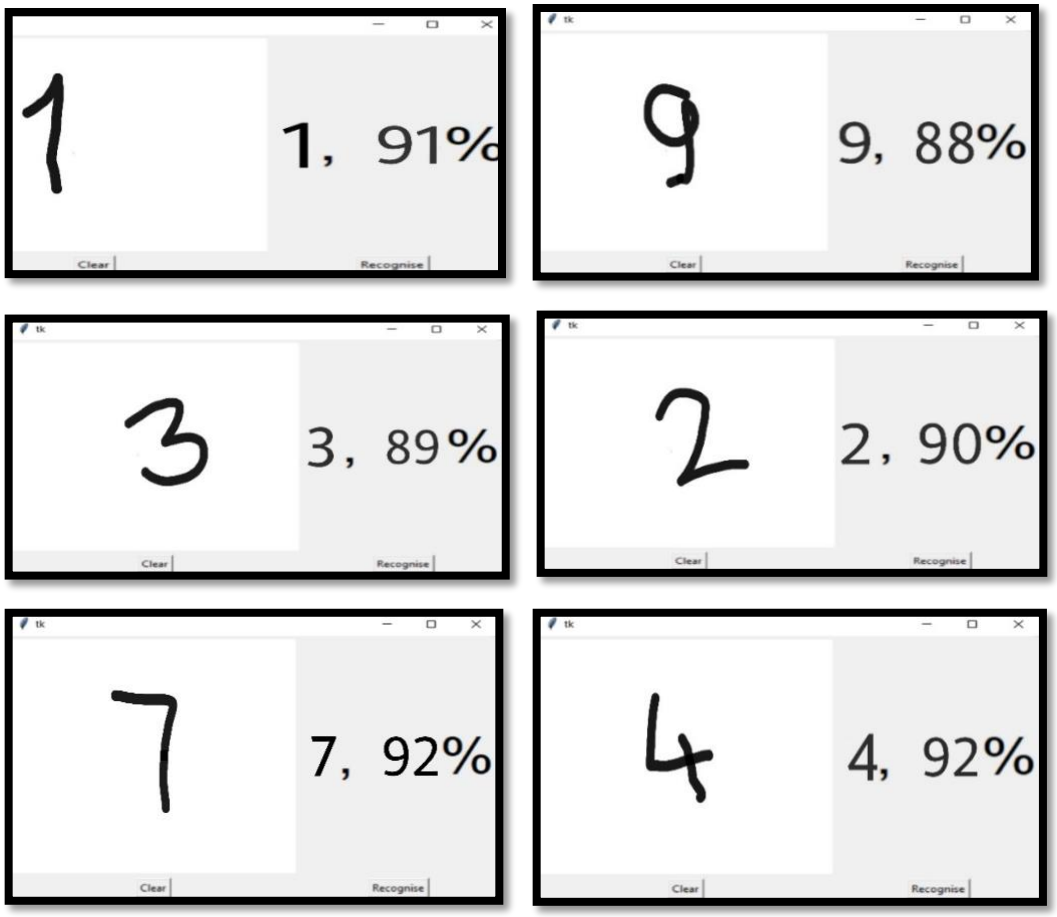


Figure 7: Identification of output

8.CONCLUSION

Here we can demonstrate a model which helps us in Identification of handwritten digits. With a few modifications this project can be extended as real time handwriting analysis and character recognition. This digit Identification is the primary step or entry to the vast field of Machine Learning, computer vision, AI. Based on the result we can conclude that CNN is now becoming one of the most efficient deep learning models than other models. This result could be more accurate if more convolution layer is attached with the training model. This digit Identification should be an excellent prototype problem for advanced techniques of deep learning technique related to neural work.

9.ACKNOWLEDGEMENT

We took opportunity to express my earnest gratitude & sincerest thanks to our departmental faculties for giving the most valuable suggestions and co-operative guidance & inspiration for this paperwork.

REFERENCES

1. Dr. Kusum Gupta 2,"A Comprehensive Review on Handwritten Digit Identification Using Various Neural Network Approaches", International Journal of Enhanced Research In Management & Computer Applications, Vol. 5, No. 5, Pp. 22-25, 2016.

2. Saeed AL-Mansouri, "Intelligent Handwritten Digit Identification using Artificial Neural Network", Int. Journal of Engineering Research and Applications, vol. 5, no. 5, pp. 46-51, 2015.

3. Ishani Patel, Vira Jagtap and Ompriya Kale. "A Survey on Feature Extraction Methods for Handwritten Digits Recognition", International Journal of Computer Applications, vol. 107, no. 12, pp. 11-17, 2014

4. Jung Kim, Xiaohui Xie, "Handwritten Identification using deep convolutional neural networks", University of California.

5. 5. Lauera, F.; Suenb, C.Y.; Blocha, G. A trainable feature extractor for handwritten digit recognition. Pattern Recognit. 2007, 40, 1816–1824.

6. Ahlawata, S.; Choudhary, A. Hybrid CNN-SVM classifier for handwritten digit recognition. Procedia Comp. Sci. 2020, 167, 2554–2560.

7. Chakraborty, S.; Paul, S.; Sarkar, R.; Nasipuri, M. Featuremap reduction in CNN for handwritten digit recognition. Recent Dev. Mach. Learn. Data Anal. 2018, 740, 143–148.

8. Hossain, M.A.; Ali, M.M. Identification of handwritten digit using convolutional neural network (CNN). Glob. J. Computer. Sci. Technol. Neural Artif. Intell. 2019, 19, 27–33.

9. Narender Kumar, Himanshu Beniwal, “Survey on Handwritten Digit Identification using Machine Learning”, H.N.B. Garhwal University, Vol-6, Special Issue-5, June (2018), ISSN: 2347-2693.

10. Pranit Patil, Bhupinder Kaur, “Handwritten Digit Identification Using Various Machine Learning Algorithms and Models”, Journal of Innovative Research in Computer Science & Technology (IJIRCST), Vol. 8, Issue- 4, July (2020), ISSN: 2347-5552.

11. V.Gopalakrishan, R. Arun, L. Sasikumar, K. Abhirami, “Handwritten Digit Identificationfor Banking System”, Kings College of Engineering, Punalkulam, Pudukottai, Journal of Engineering Research & Technology (IJERT), (2021), ISSN: 2278-0181.

12. GuyonI Poujaud LPersonnaz GDreyfus JDenkerand YLeCun Comparing Dierent Neural Net Architectures for Classifying Handwritten Digitsin Proc IJCNN II Washington DC IEEE.

13. Bushra Aurangzeb, Memuna Malik, “A Survey on Handwritten Digit Recognition”, COMSATS Institute of Information technology , Vol. 4, Issue 2, April (2015), ISSN: 2306-708X.

14. M. D. Zeiler and R. Fergus, “Visualizing and understanding convolutional neural networks,” arXiv: 1311.2901, 2013.

15. Fabien Lauer, Ching Y. Suen, and G´erard Bloch “A trainable feature extractor for handwritten digit recognition”, Journal Pattern Recognition, Elsevier, 40 (6), pp.1816-1824, 2007.

16. Saeed AL-Mansoori, "Intelligent Handwritten Digit Recognition using Artificial Neural Network", Int. Journal of Engineering Research and Applications, vol. 5, no. 5, pp. 46-51, 2015.

17. derektanderson.com/FuzzyLibrary/customnetmnst.h tml.

18. Huimin Wu. CNN-Based Recognition of Handwritten Digits in MNIST Database.

19. Laith Alzubaidi & Jinglan Zhang, “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions” arXiv: 10.1186/s40537-021-00444-8.

11. BIOGRAPHIES

Mr. Pranab Hazra is working as Assistant Professor in the department of Electronics and Communication Engineering at Narula Institute of Technology. His research interest in Biomedical signal processing, Sferic signal analysis, remote sensing, and GIS. He has 15 number of Scopus Indexed journal and 12 number of WOS Publications. He also has 3 book chapters as well as he is one of the editors of the book namely “Advanced Computation on Electronics and Remote Sensing”. He has published 3 patents. He has many students’ publication also. He was also worked as Sr.Engineer at Fibcom India Ltd., Gurgaon. He was achieved the **Best Paper Award** in **ICCACCS-2018.**

Mr. MAINAK DEY, a 3rd year (2024 pass out) student of B-TECH in ECE Department of Narula Institute of Technology an Autonomous Institute Affiliated to Makaut.

Mr. SAYANDIP KUMAR, a 3rd year (2024 pass out) student of B-TECH in ECE Department of Narula Institute of Technology an Autonomous Institute Affiliated to Makaut.

Miss. ANURUPA PAUL, a 3rd year (2024 pass out) student of B-TECH in ECE Department of Narula Institute of Technology an Autonomous Institute Affiliated to Makaut

Miss. KAUSHIKI MITRA, a 3rd year (2024 pass out) student of B-TECH in ECE Department of Narula Institute of Technology an Autonomous Institute Affiliated to Makaut.