DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

BANGLADESH ARMY UNIVERSITY OF SCIENCE & TECHNOLOGY (BAUST)

SAIDPUR CANTONMENT, NILPHAMARI

(Project Proposal)

Course Code: CSE 4132 **Course Title:** Artificial Neural Networks and Fuzzy System Sessional

Date: May 02, 2023

1. **Name of the Students (with ID) :**
   - Jarine Reshni Nijhum (200101026)
   - Md Ali Arman (200101056)
   - Maisha Maliha Jahan (200101060)
   - Md Wahid Rahman (200101063)

2. **Present Address** **:** Shohid Dr. Zikrul Haque Hall, Bir Protiki Taramon Bibi Hall, Bangladesh Army University of Science & Technology (BAUST), Saidpur Cantonment, Saidpur, Nilphamari.

3. **Name of the Department :** Computer Science & Engineering

   Program **:** Bachelor of Science in Computer Science and Engineering

4. **Tentative Title** **:** Handwritten Digit Classification System using Machine Learning.

5. **Introduction**

   Handwritten digit recognition is the ability of a computer to recognize the human handwritten digits from different sources like images, papers, touch screens etc. and classify them into 10 predefined classes (0-9). The implementation of handwritten digit recognition by Convolutional Neural Network is done using Keras. It is an open-source neural network library that is used to design and implement deep learning models. From Keras, we have used a Sequential class which allowed us to create model layer-by-layer. Handwritten digit recognition is the process to provide the ability to machines to recognize human handwritten digits. It is not an easy task for the machine because handwritten digits are not perfect, vary from person-to-person, and can be made with many different flavors. The idea is to take a large number of handwritten digits, known as training examples, and then develop a system which can learn from those training examples. In other words, the neural network uses the examples to automatically infer rules for recognizing handwritten digits. Most people effortlessly recognize those digits as 504192. That ease is deceptive. In each hemisphere of our brain, humans have a primary visual cortex, also known as V1, containing 140 million neurons, with tens of billions of connections between them **[1]**. And yet human vision involves not just V1, but an entire series of visual cortices - V2, V3, V4, and V5 doing progressively more complex image processing **[1].** We carry in our heads a supercomputer, tuned by evolution over hundreds of millions of years, and superbly adapted to understand the visual world. Recognizing handwritten digits isn't easy. Rather, we humans are stupendously, astoundingly good at making sense of what our eyes show us. But nearly all that work is done unconsciously and so we don't usually appreciate how tough a problem our visual systems solve. The idea is to take a large number of handwritten digits, known as training examples and then develop a system which can learn from those training examples.

Figure-01: Handwritten training set

In other words, the neural network uses the examples to automatically infer rules for recognizing handwritten digits.

6. Background and Present State of the Problem

There are a number of ways and algorithms to recognize handwritten digits, including Deep Learning/CNN, SVM, Gaussian Naive Bayes, KNN, Decision Trees, Random Forests **[3]**. Most algorithms for segmenting connected handwritten digit strings are based on the analysis of the foreground pixel distributions and the features on the upper/lower contours of the image. In this paper, a new approach is presented to segment connected handwritten two-digit strings based on the thinning of background regions. The algorithm first locates several feature points on the background skeleton of a digit image. Possible segmentation paths are then constructed by matching these feature points. With geometric property measures, all the possible segmentation paths are ranked using fuzzy rules generated from a decision-tree approach. Finally, the top ranked segmentation paths are tested one by one by an optimized nearest neighbor classifier until one of these candidates is accepted based on an acceptance criterion. Experimental results on NIST special database 3 show that our approach can achieve a correct classification rate of 92.5% with only 4.7% of digit strings rejected **[4]**, which compares favorably with the other techniques tested. The MNIST handwritten digit classification problem is a standard dataset used in computer vision and deep learning. Although the dataset is effectively solved, it can be used as the basis for learning and practicing how to develop, evaluate, and use convolutional deep learning neural networks for image classification from scratch. This includes how to develop a robust test harness for estimating the performance of the model, how to explore improvements to the model and how to save the model and later load it to make predictions on new data.

7. Objective with Specific Aims and Possible Outcome

The objective of this work is to develop a machine learning model that can accurately recognize handwritten numbers. The objective with specific aims of this project are as follows:

- Accurately recognising images of Handwritten digits based on classification methods for multivariate data.

- To explore the capabilities of machine learning in solving pattern recognition and image processing problems and developing new machine learning algorithms.

- Enhancing automation which is needed in postal address recognition, form filling, and signature verification to improve the efficiency and to achieve high accuracy on the test set by training the model on a large dataset of handwritten digit images.

The possible outcome of this project is creating a system that can accurately recognize handwritten digits in real-time. And analyzing of the strengths and weaknesses of the model, as well as its potential applications in areas such as character recognition and digit classification.

8. Outline of Methodology Design

The methodology for this project involves the following steps:

1. Data Collection: The first step is to gather a dataset of handwritten digits. We will create our own dataset, which could be more relevant to our use case.

2. Data Preprocessing: Next step is data preprocessing. Normalization, resizing, and feature extraction are instances of this. Normalization ensures that incoming data has the same scale to make machine learning easier. Resizing images to standardize image sizes. Feature extraction uses edges and corners to help the machine learning system distinguish between digits.

3. Feature Extraction: Feature extraction is the process of selecting and extracting important features from the input data, such as pixel intensity, edges, texture, or shape. These features are used by the machine learning model to distinguish between different digits in handwritten digit recognition. Common techniques for feature extraction include pixel intensity, edge detection, histogram of oriented gradients (HOG), local binary patterns (LBP), and scale-invariant feature transform (SIFT). **[2]**

4. Model Selection: We will select four types of models and they are - ResNet50, Xception net, Inception net V3, VGG net16. Each model has its own advantages and disadvantages and the choice of model will depend on factors such as the size of the dataset, the complexity of the problem, and the available computing resources.

5. Model Training: The next step is to train the selected model on the preprocessed data. This is done by feeding the model its training data and tweaking its parameters until the difference between the expected and actual outputs is as small as possible. And we will apply three types of training (Baseline, Transfer Learning and Fine Tuning) on each of the models to evaluate the four distinct kinds of models side by side and select the one that best meets our criteria.

6. Model Evaluation: Once the model is trained, it is evaluated on a separate testing dataset to assess its performance. The evaluation metrics could include accuracy, precision, recall, and F1 score. The model's performance can be further analyzed by generating a confusion matrix and plotting a receiver operating characteristic (ROC) curve.

In summary, developing a handwritten digit recognition system using machine learning involves data collection, preprocessing, feature extraction, model selection, model training, model evaluation. Following this methodology design should result in a well-performing system that can accurately recognize hand
written digits.

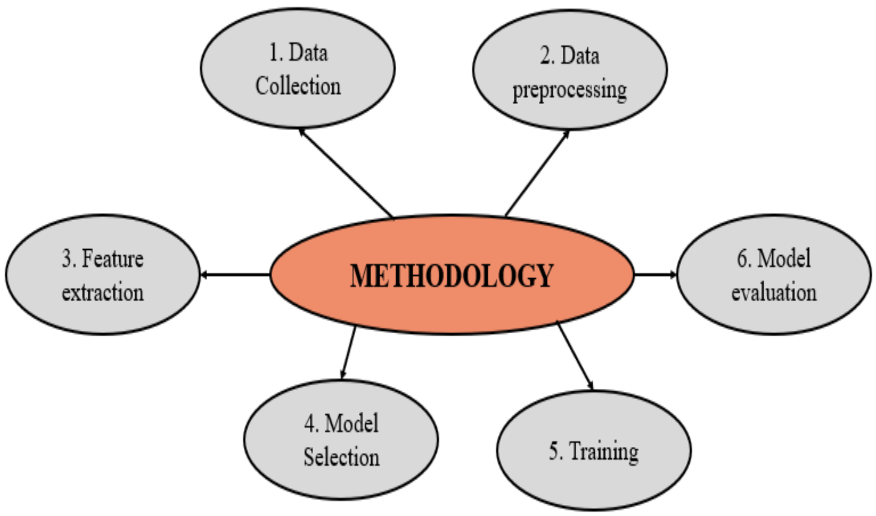

Figure-02: Outline of Methodology Design

9. Resources Required to Accomplish the Task

- Python
- Google Colab
- TensorFlow
- Keras
- Scikit-learn
- OpenCV
- PyTorch,
- Jupiter Notebook

10. References

[1] Aybars, Ozan. *Using neural nets to recognize handwritten digits*. Diss. Zakład Systemów Przetwarzania Informacji, 2021.

[2] LeCun, Yann, et al. "Learning algorithms for classification: A comparison on handwritten digit recognition." Neural networks: the statistical mechanics perspective 261.276 (1995): 2.

[3] Pashine, Samay, Ritik Dixit, and Rishika Kushwah. "Handwritten digit recognition using machine and deep learning algorithms." arXiv preprint arXiv:2106.12614 (2021).

[4] Lu, Zhongkang, et al. "A background-thinning-based approach for separating and recognizing connected handwritten digit strings." Pattern Recognition 32.6 (1999): 921-933.

[5] Kussul, Ernst, and Tatiana Baidyk. "Improved method of handwritten digit recognition tested on MNIST database." Image and Vision Computing 22.12 (2004): 971-981.

Name and ID of the Students	**Signature of the Students**
Jarine Reshni Nijhum 200101026	
Md Ali Arman 200101056	
Maisha Maliha Jahan 200101060	
Md Wahid Rahman 200101063	