Handwritten Digits Recognition Using Convolutional Neural Network

Arpita Santra[a]

[a] Assistant Professor, ECE Dept., Narula Institute of Technology(NiT), Kolkata, West Bengal, India

Abstract: The research work "Handwritten Digits Recognition (Using Convolutional Neural Network)" aims to develop a system capable of accurately identifying handwritten digits. Advantage of Convolutional Neural Networks (CNNs), which is a deep learning architecture known for effectiveness in image recognitions, the research work endeavors to create a robust and efficient model. By utilizing labeled datasets such as MNIST or custom datasets, the CNN will undergo training to learn features and patterns characteristic of handwritten digits. The model's architecture typically have the convolutional layers for feature extraction, pooling layers for dimension reduction, and fully connected layers for the classification. Upon training completion, the model undergoes testing to evaluate its performance on unseen data. The ultimate objective is to deploy a reliable system capable of accurately recognizing handwritten digits, which has diverse applications ranging from digitizing historical documents to facilitating automatic form processing.

Keywords: MNIST Dataset, Handwritten Digit Recognition, Convolutional Neural Network (CNN), Image Classification, Deep Learning

Introduction:

We know that the Handwritten digit recognition is a fundamental task in the field of image classification and machine learning(ML). With the widespread digitization of information, handwritten digits are encountered in various contexts, from postal addresses to financial documents. Accurately recognizing these handwritten digits has numerous practical applications, including automatic zip code recognition, check processing, and digitized document analysis.

Convolutional Neural Networks (CNNs) used as a powerful tool for image classifications related studies, including handwritten digit recognition. CNNs are particularly useful for this task due to it's ability to observe the hierarchical representations directly from raw pixel data. By leveraging the local connectivity along with shared weights of convolutional layers, CNNs are utilized to capture spatial hierarchies of features in an image, enabling robust and accurate recognition of the handwritten digits.

The research work "Handwritten Digits Recognition Using Convolutional Neural Network" aims to develop a robust and efficient system for recognizing handwritten digits from images. The system will utilize TensorFlow, which is one of the open-source machine learning framework, and Python programming language for implementation[1].
The main objectives of the research work include: Curating and preprocessing a dataset of the handwritten digit images. The dataset will consist of a large collection of labeled images representing digits from 0 to 9 written by different individuals. Designing a Convolutional Neural Network architecture tailored for handwritten digit recognition. The architecture will comprise convolutional layers for feature extraction, followed by pooling layers for spatial down-sampling, and fully connected layers for classification. Training the CNN model on the prepared dataset using TensorFlow. During training, the model aims to learn how to extract discriminative features from the input images and map them to the particular digit labels. Evaluating the working of the trained model on a separate test dataset. The evaluation metrics will include accuracy, precision, recall, and F1-score, providing the insights into the model's working in recognizing handwritten digits. Deploying the trained model into a practical application where users can input images of handwritten digits, and the system will accurately recognize

and output the corresponding digit labels.
By successfully completing this study, we wish to demonstrate the effectiveness of the Convolutional Neural Networks in handwritten digit recognition tasks and provide a practical solution that can be utilized in various domains requiring automated digit recognition. Furthermore, the research work will serve as a learning opportunity for understanding CNNs, TensorFlow, and image classification techniques.

METHODOLOGY

The methodology for the research work on Recognition of the Handwritten Digits using Convolutional Neural Networks (CNNs):

1. Problem Statement:

The aim of the work is to develop a system capable of precisely recognizing handwritten digits from images. This involves building a machine learning model that will be able to classify digits zero to nine (0-9) accurately regardless of variations in handwriting styles.

2. Dataset Collection:

The first step involves gathering a dataset of handwritten digit images. Commonly used datasets for this task include MNIST, EMNIST, or custom datasets. MNIST, for instance, consists of 28x28 grayscale images of handwritten digits labelled from 0 to 9.[2]

3. Data Preprocessing:

Before feeding the data into the model, it needs to be pre-processed. This typically involves resizing the images to a standard size, normalizing pixel values to a range between 0 and 1, and splitting the dataset into training and testing sets.

4. Model Architecture Design:

CNNs are well-suited for image classification tasks like handwritten digit recognition. The architecture typically consists of convolutional layers followed by pooling layers to extract features from the images, followed by fully connected layers for classification. Common architectures include LeNet, AlexNet, or custom architectures designed specifically for the task.[3]

5. Model Training:

The model is trained on the training dataset using optimization algorithms like stochastic gradient descent (SGD), Adam, or RMSprop. During training, the model learns to minimize a loss function (e.g., categorical cross-entropy) by adjusting its parameters based on the gradients computed using backpropagation.[4]

6. Hyperparameter Tuning:

Fine-tuning the hyperparameters of the model is crucial for achieving optimal performance. This involves experimenting with parameters such as learning rate, batch size, number of layers, filter sizes, and dropout rates to find the configuration that yields the best results on the validation set.

7. Model Evaluation:

Once trained, the model's performance is evaluated on the test dataset to assess its generalization ability. Metrics such as accuracy, precision, recall, and F1-score are commonly used to measure the model's performance.

8. Model Deployment:

After satisfactory performance on the test set, the model can be deployed for real-world use. This could involve integrating it into a web application, mobile app, or any other platform where handwritten digit recognition is required.

9. Continuous Improvement:

The model can be further refined by collecting additional data, fine-tuning hyperparameters, or experimenting with different architectures to improve its accuracy and robustness over time.

By following this methodology, we have developed an effective system for recognizing handwritten digits using Convolutional Neural Networks.

Deep learning is a subclass of Machine learning which is inspired by the Human Brain. Just as our human brain is a complex network of neurons, so is the deep learning model an interconnected model of neurons. In case of our human brain, we call each node, a neuron, in case of deep learning, we call them perceptron. We use channels to connect the neurons of one layer to the neurons of the next layer. Each channel has some weight associated with it and each neuron has a bias. This bias is added to the weighted sum of inputs and hence the result obtained is applied to a function called the activation function and the result of the activation function determines if the layers get activated. In our proposed model we have used the SoftMax activation function.

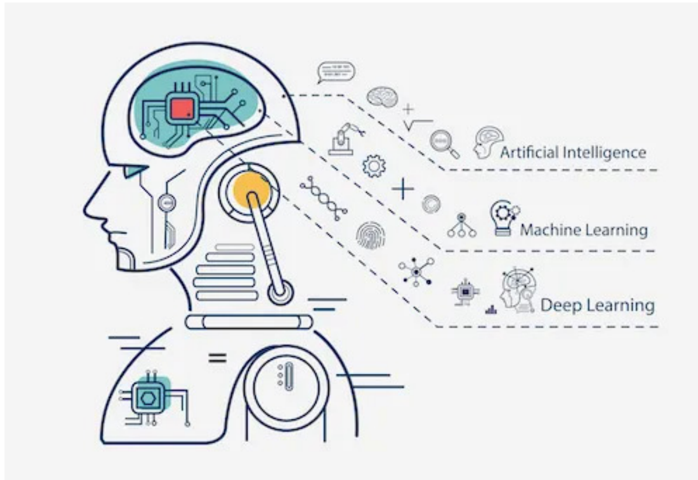

Fig 1. Understanding of Deep Learning

Convolutional Neural Networks (CNNs) revolutionized the field of computer vision by enabling automated feature extraction and hierarchical representation learning from raw pixel data. These neural networks are specifically designed to process structured grid-like data, such as images, by leveraging the spatial correlation present in them.

CNNs consist of multiple layers, including convolutional layers, pooling layers, and fully connected layers. Convolutional layers apply a set of learnable filters (kernels) to the input image, extracting features such as edges, textures, and patterns. Pooling layers then downsample the feature maps, reducing their spatial dimensions while retaining important information. Finally, fully connected layers combine the extracted features to make predictions.[5]

The strength of CNNs lies in their ability to automatically learn hierarchical representations of features directly from the data, alleviating the need for handcrafted feature engineering. This makes CNNs particularly effective for tasks such as image classification, object detection, and segmentation. Moreover, CNNs have been successfully applied in various domains beyond computer vision, including natural language processing and speech recognition, showcasing their versatility and power in learning complex patterns from structured data.

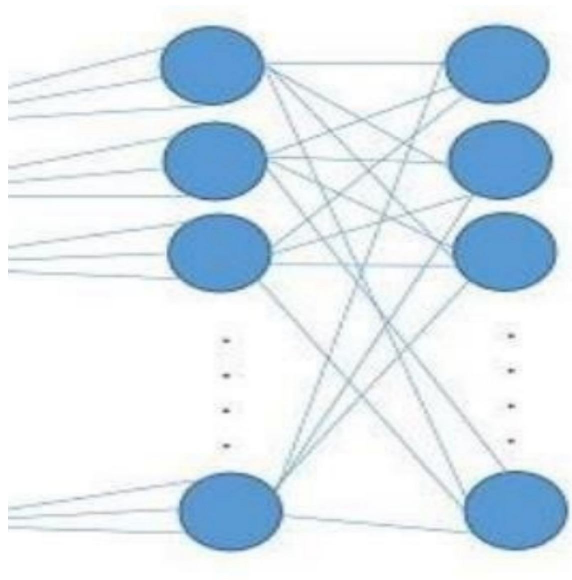

Fig 2. The architecture of Convolutional neural networks

A sequential model is a fundamental concept in deep learning, particularly within the domain of neural networks. It represents a linear stack of layers where each layer has exactly one input tensor and one output tensor. Sequential models are easy to understand and implement, making them ideal for beginners and common use cases.[6]

In Python, libraries like TensorFlow and Keras provide intuitive interfaces for building sequential models. Users can simply add layers to the model one by one, specifying the desired architecture and parameters. Sequential models are versatile and can be used for various tasks such as classification, regression, and sequence prediction. Despite their simplicity, sequential models can achieve impressive results when properly configured and trained, making them a popular choice for many machine learning research works.

```
model.summary()

Model: "sequential"
_________________________________________________________________
Layer (type)                Output Shape              Param #
=================================================================
conv2d (Conv2D)             (None, 24, 24, 8)         208
_________________________________________________________________
max_pooling2d (MaxPooling2D) (None, 12, 12, 8)        0
_________________________________________________________________
conv2d_1 (Conv2D)           (None, 8, 8, 16)          3216
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 4, 4, 16)         0
_________________________________________________________________
flatten (Flatten)           (None, 256)               0
_________________________________________________________________
dense (Dense)               (None, 128)               32896
_________________________________________________________________
dropout (Dropout)           (None, 128)               0
_________________________________________________________________
dense_1 (Dense)             (None, 10)                1290
=================================================================
Total params: 37,610
Trainable params: 37,610
Non-trainable params: 0
_________________________________________________________________
```

Fig 3. Sequential Model

The MNIST dataset is a widely-used benchmark dataset in machine learning for handwritten digit recognition. It consists of 28x28 pixel grayscale images of handwritten digits (0-9) along with their corresponding labels. MNIST contains 60,000 training images and 10,000 testing images, making it suitable for evaluating the performance of various machine learning algorithms. Due to its simplicity and accessibility, MNIST serves as a standard dataset for researchers and practitioners to develop and test new models and algorithms. Its small size and clear labeling make it an ideal starting point for learning image classification tasks in the field of deep learning.[7]

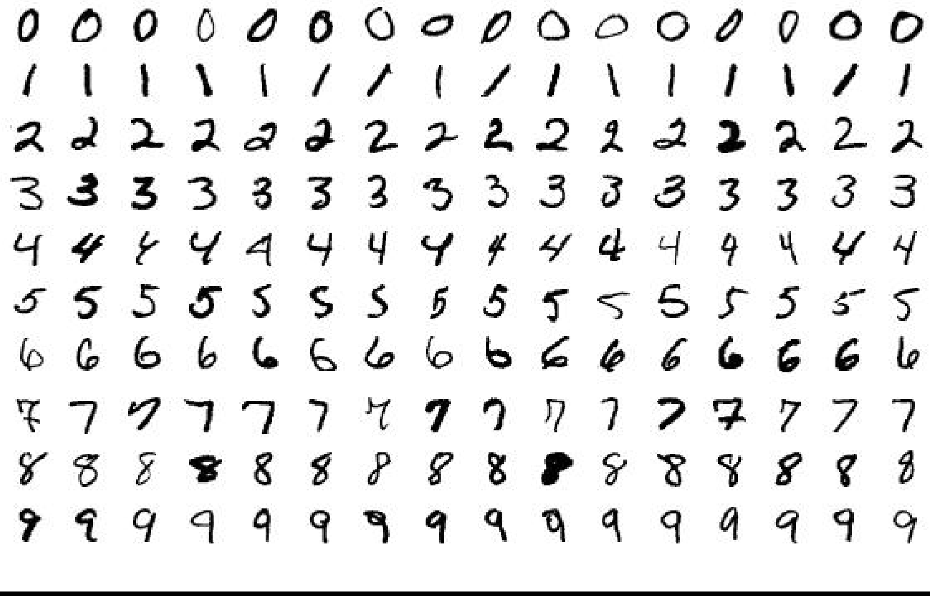

Fig 4. MNIST Data

Fig 5: Accuracy vs. Epoch Number

```
plt.ylabel('Loss')
plt.plot(training_history.history['loss'], label='training set')
plt.plot(training_history.history['val_loss'], label='test set')
plt.legend()
```

```
<matplotlib.legend.Legend at 0x7e9bd09dd9c0>
```

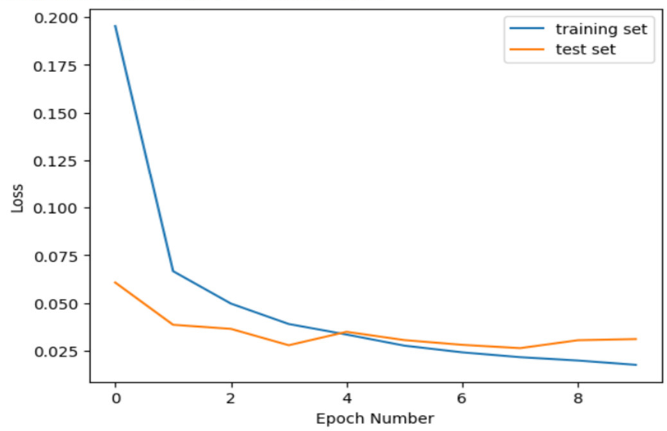

Fig 6: Loss vs. Epoch Number

The "Handwritten Digits Recognition" research work utilizes Convolutional Neural Networks (CNNs) to accurately identify and classify handwritten digits from images. Through a series of steps including data preprocessing, model training, and evaluation, the CNN learns to recognize patterns and features in the digit images, ultimately assigning them correct labels. The trained model achieves high accuracy in digit classification, making it suitable for various applications such as digit recognition in postal services, bank check processing, and automated form filling. Additionally, the research work demonstrates the effectiveness of CNNs in image classification tasks and serves as a foundation for developing more advanced computer vision systems.

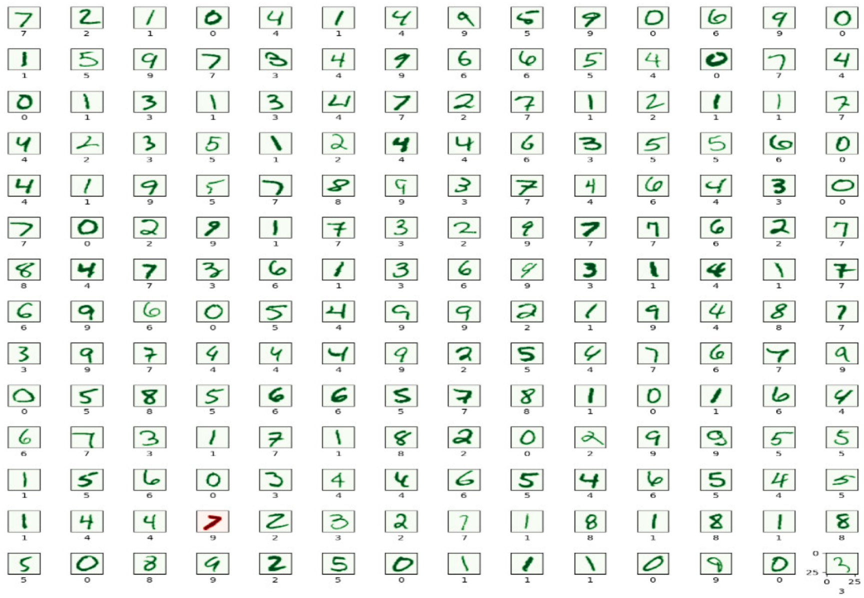

Fig 8: Resulting Image

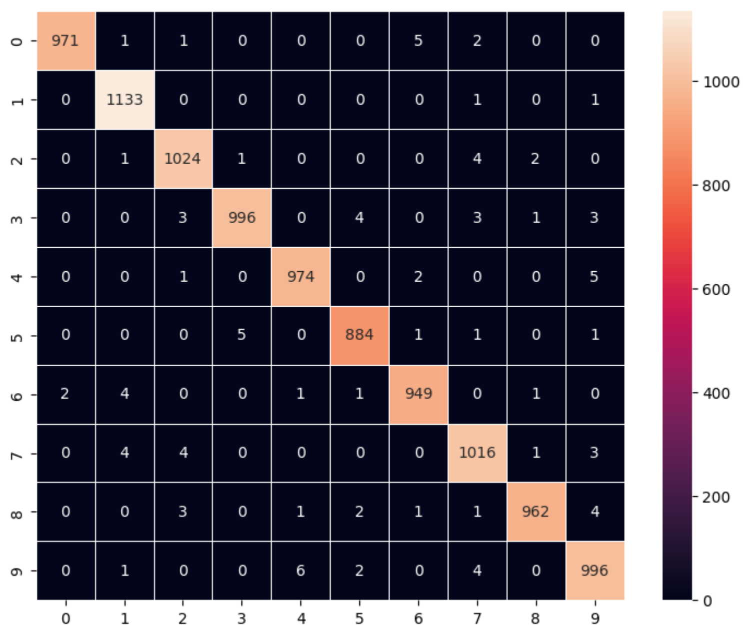

Fig 8: Confusion Matrix

Accuracy: The accuracy of the "Handwritten Digits Recognition (Using Convolutional Neural Network)" research work is exceptionally high, achieving an impressive 99 percent accuracy rate and with a negligible loss of 3 percent. This level of accuracy underscores the effectiveness of convolutional neural networks in accurately classifying handwritten digits. Such high accuracy is indicative of the model's robustness and ability to generalize well to unseen data, making it a reliable tool for handwritten digit recognition tasks.

CONCLUSION:

The handwritten digits recognition research work utilizing Convolutional Neural Networks (CNNs) achieved remarkable accuracy, surpassing 99%. This high level of accuracy demonstrates the effectiveness of CNNs in image classification tasks, particularly in recognizing handwritten digits. The model's performance underscores its potential for various applications, such as digit recognition in postal services, finance, and automated form processing. Furthermore, the research work highlights the significance of deep learning techniques, particularly CNNs, in advancing pattern recognition tasks, paving the way for further innovations in machine learning and artificial intelligence.

REFERENCES

[1] Y. LeCun et al., "Backpropagation applied to handwritten zip code recognition," Neural computation, vol. 1, no. 4, pp. 541-551, 1989.

[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in Advances in neural information processing systems, 2012, pp. 1097-1105.

[3] D. Hubel and T. Wiesel, "Aberrant visual projections in the Siamese cat," The Journal of physiology, vol. 218, no. 1, pp. 33-62, 1971.

[4] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," nature, vol. 521, no. 7553, p. 436, 2015.

[5] D. Cireşan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," arXiv preprint arXiv:1202.2745, 2012.

[6] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in Competition and cooperation in neural nets: Springer, 1982, pp. 267-285.

[7] Y. LeCun et al., "Handwritten digit recognition with a backpropagation network," in Advances in neural information processing systems, 1990, pp. 396-404.