**Handwritten Digit Recognition**

**A report on**

**Machine Learning Lab Project**

**[CSE-3181]**

Submitted By

**Aditya Nerusu (210962196)**

**Rinchen Norbu (210962210)**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGGINEERING**

**MANIPAL INSTITUTRE OF TECHNOLOGY,**

**MANIPAL ACADEMY OF HIGHER EDUCATION**

November 2023

**Handwritten Digit Recognition**

Rinchen Norbu (210962210)

Aditya Nerusu (210962196)

Department of Computer Science and Engineering,

Manipal Institute of Technology

Manipal Academy of Higher Education, India

[rinchennorbu441@gmail.com](mailto:rinchennorbu441@gmail.com)

adityaadi2744@gmail.com

***Abstract— The "Handwritten Digit Recognition using Machine Learning and Artificial Neural Networks" project stands as an innovative exploration at the intersection of machine learning and pattern recognition. The primary aim of this project is to develop a robust system proficient in accurately identifying handwritten digits from scanned images through the application of Artificial Neural Networks (ANNs) and machine learning techniques.***

***The project's framework relies on the utilization of ANNs, particularly deep learning architectures like convolutional neural networks (CNNs) and recurrent neural networks (RNNs). By leveraging these networks' ability to learn intricate patterns and features, the project aims to achieve superior accuracy in recognizing and classifying handwritten digits across diverse styles and complexities.***

***Through meticulous data preprocessing, feature extraction, and model optimization, this project endeavors to push the boundaries of handwritten digit recognition accuracy. The implications of this research span various domains, from enhancing Optical Character Recognition (OCR) systems to advancing automation in digitized data analysis.***

***This report provides a comprehensive overview of the methodologies, experiments, findings, and potential implications of the project. It emphasizes the role of ANNs and machine learning in transforming handwritten data into digital information, contributing significantly to the field of pattern recognition and machine intelligence.***

***Keywords— Handwritten Digit Recognition, Machine Learning, Artificial Neural Networks, Convolutional Neural Networks, Recurrent Neural Networks, Pattern Recognition, Optical Character Recognition.***

**INTRODUCTION**

In the annals of machine learning, the pursuit of Handwritten Digit Recognition has been an enduring quest, tracing its roots back to the advent of pattern recognition and artificial intelligence. This endeavor finds its genesis in the early stages of computing, where researchers grappled with the challenge of automating the recognition of handwritten characters.

The history of Handwritten Digit Detection can be traced to pioneering efforts that sought to decode and interpret handwritten content. Early methods relied heavily on feature engineering and rule-based systems, where handcrafted features were extracted to distinguish between different digits. However, these approaches were limited by the variability and complexity inherent in human handwriting.

The watershed moment arrived with the introduction of machine learning techniques, particularly the utilization of Neural Networks, which revolutionized the landscape of digit recognition. The MNIST dataset, a collection of handwritten digits, became a cornerstone in benchmarking and evaluating the efficacy of various recognition algorithms. The advent of Neural Networks, especially deep learning architectures like Convolutional Neural Networks (CNNs), significantly bolstered the accuracy and robustness of digit recognition systems. As this field evolved, researchers and practitioners continuously refined algorithms, exploring novel methodologies and architectures to improve recognition accuracy and scalability. The transition towards deep learning marked a pivotal shift, enabling systems to learn intricate patterns and representations directly from the data, transcending the limitations of handcrafted features.

Our project stands on the shoulders of this rich history, leveraging advancements in machine learning and Neural Networks to further the frontiers of Handwritten Digit Recognition. Beyond the historical milestones, our endeavor incorporates modern techniques, culminating in a system that not only detects scanned images of handwritten digits but also integrates a user-friendly interface for real-time digit recognition and interaction.

By building upon this historical trajectory, our project signifies not just an evolution but a convergence of traditional handwritten digit recognition with cutting-edge technologies, embodying the transformative journey from the early days of pattern recognition to the era of intelligent digit interpretation and interaction.

**EXISTING LITERATURE REVIEW**

In the realm of machine learning and computer vision, Handwritten Digit Recognition stands as a testament to the evolution of pattern recognition and artificial intelligence. The journey traces back to the early days of computer science, marked by pioneering attempts to automate the recognition of handwritten characters and digits.

The inception of this quest dates to the mid-20th century when researchers embarked on the formidable task of decoding handwritten content. Initial attempts relied on rudimentary techniques like template matching and simple pattern recognition algorithms. However, these approaches were constrained by their inability to handle the diverse variations and intricacies inherent in handwritten digits. Subsequent decades saw significant strides in pattern recognition and machine learning methodologies. In the late 1960s and 1970s, advancements in feature extraction techniques emerged, focusing on identifying specific characteristics of handwritten digits. These efforts paved the way for more sophisticated algorithms that aimed to capture strokes, curvature, and other defining attributes. The breakthrough arrived with the advent of Neural Networks in the 1980s and 1990s, albeit their limited depth and computational capabilities constrained their performance. However, these early neural networks laid the foundation for future advancements in digit recognition.

A pivotal moment in the history of Handwritten Digit Recognition came with the introduction of benchmark datasets like MNIST in the late 1990s. MNIST's availability as a standardized dataset enabled researchers to benchmark and compare various recognition algorithms, fueling advancements in the field. The MNIST database (Modified National Institute of Standards and Technology database) is a large database of handwritten digits that is commonly used for training various image processing systems.

This project stands at the nexus of this historical trajectory, leveraging insights from the evolution of pattern recognition methodologies and the integration of user-friendly interfaces. By amalgamating these historical advancements, this project aims not only to detect handwritten digits with precision but also to pioneer intuitive, real-time interaction through an integrated Graphical User Interface (GUI).

**METHODOLOGY**

1. Data Gathering

* The very first step in the methodology of our project is to acquire the image from which to extract the digits. For this we will use the MNIST dataset .
* The MNIST database contains 60,000 training images and 10,000 testing images.
* Half of the training set and half of the test set were taken from NIST's training dataset, while the other half of the training set and the other half of the test set were taken from NIST's testing dataset.

1. Preprocessing

* After collecting the data, the next step was data preprocessing. The collected images were preprocessed to make it easier for our main code to identify important features in.
* The original dataset was a set of 128x128 binary images, processed into 28x28 grayscale images. There were originally 60k samples in both the training set and the testing set, but 50k of the testing set were discarded.

1. Nural network

* Our implementation involves a Neural Network architecture consisting of a single hidden layer housing 100 activation units (excluding bias units). The dataset is loaded from a .mat file, with features (X) and labels (y) extracted. To ensure computational stability and avoid overflow, the features are rescaled by dividing them by 255, effectively scaling them within the range of [0,1].
* Data segmentation involves splitting it into distinct sets for training and testing, comprising 60,000 and 10,000 examples, respectively. The training process commences with feedforward propagation on the training set, computing the hypothesis, followed by backpropagation to minimize the error across network layers.
* To mitigate overfitting, a regularization parameter lambda is introduced and set to 0.1, optimizing the model's generalization capabilities. The optimization procedure involves running the optimizer for 70 iterations, seeking the most suitable model fit.
* We will utilize a Neural Network architecture with one hidden layer containing 100 activation units, excluding bias units, designed for digit recognition tasks.
* And we will initialization of Thetas (weights) within a certain range (-0.15 to +0.15) to break symmetry and enhance model convergence.

1. Data Preparation and Segmentation:

* Data is sourced from the MNIST dataset, which is then preprocessed by scaling the features (X) within the range [0,1] to facilitate numerical stability and prevent computational overflow.
* Segmentation of the dataset into a training subset of 60,000 examples and a separate testing subset of 10,000 examples.

1. Training and Optimization:

* Forward propagation on the training set to compute the hypothesis followed by backpropagation to minimize inter-layer errors, facilitated by the optimization process.
* Introduction of a regularization parameter (lambda\_reg = 0.1) to counter overfitting, thereby enhancing the model's generalization capacity.

1. Model Validation and Metrics:

* Assessment of model performance on the test set to ascertain accuracy.
* Calculation of precision metrics like training set accuracy and precision, providing insights into the model's predictive capabilities and precision rates.

1. Save Optimized Parameters:

* Preservation of the trained model's optimized parameters (Theta1 and Theta2) into .txt files for future reference or utilization.

1. Graphical User Interface (GUI) Implementation:

* Creation of a GUI interface using Tkinter for handwriting digit recognition.
* Enable users to draw digits on the GUI canvas and capture these images.
* Image processing involves converting the captured image to grayscale and resizing it to the required (28 X 28) pixel size for digit recognition.

1. Digit Recognition and Display:

* Utilize the previously trained model's parameters (Theta1 and Theta2) to predict the drawn digit.
* Display the recognized digit on the GUI interface, providing users with real-time digit recognition capabilities.

1. Digit Recognition and Display:

* Utilize the previously trained model's parameters (Theta1 and Theta2) to predict the drawn digit.
* Display the recognized digit on the GUI interface, providing users with real-time digit recognition capabilities.

**RESULTS AND DISCUSSION**

The output from the code execution showcases the iterative progression and optimization process of the neural network model using the L-BFGS-B optimization method. It provides valuable insights into the model's convergence, function value minimization, and key performance metrics.

A white number on an orange square

Description automatically generated

**Convergence and Optimization:**

The optimization process initiated with an initial function value of approximately 6.35. Iteratively, the function value (F) decreased consistently across iterations, converging to approximately 0.07 within 100 iterations. This reduction in the function value signifies the model's convergence and improvement in minimizing the cost function.

**Optimization Algorithm Details:**

The algorithm utilized the L-BFGS-B optimization method, performing a total of 105 function evaluations across 100 iterations. It explored one segment during Cauchy searches, highlighting the efficiency of the optimization method in this context.

**Results Metrics:**

Test Set Accuracy: The model achieved an accuracy of approximately 97.44% on the test dataset, showcasing its ability to generalize well on unseen data.

Training Set Accuracy: The accuracy attained on the training set stands at approximately 99.47%, indicating that the model effectively learned patterns within the training data.

Precision: Precision, a measure of the model's exactness in prediction, was calculated at approximately 99.47%, showcasing the high precision of the model's predictions.

Limitation:

The optimization process reached the total number of iterations, suggesting that further improvement or refinement might have been achievable with additional iterations. The stoppage due to reaching the total iteration limit indicates potential areas for further exploration or optimization adjustments.

Overall Assessment:

The model demonstrates promising performance with high accuracies on both training and test datasets. Its precision indicates strong predictive capability, showcasing a high level of confidence in the model's predictions. However, there might be scope for fine-tuning the model further to potentially enhance its performance or efficiency.

**CONCLUSION**

In the exploration of handwritten digit recognition, our journey has been a quest to unravel the intricacies of machine learning and neural networks applied to this specific domain. Our project sought to tackle the challenge of accurately identifying and classifying handwritten digits, a task with far-reaching implications in various fields, including automation, digitization, and pattern recognition.

At its core, our research delved into the development and optimization of a neural network model tailored for handwritten digit recognition. Through meticulous training, fine-tuning, and utilization of the L-BFGS-B optimization method, we endeavored to create a robust system capable of accurately identifying handwritten digits.

Our code implementation depicted a vivid picture of the optimization process, showcasing the iterative convergence of the model and its subsequent performance metrics. With an iterative decrease in the cost function and high accuracy rates on both training and test datasets, our model exhibited strong potential for accurately recognizing handwritten digits.

The implications of our work are far-reaching. Handwritten digit recognition stands as a foundational element in numerous applications, such as character recognition systems, automated form processing, and data digitization. The high precision achieved by our model underscores its potential in facilitating seamless automation and accurate data extraction from handwritten documents.

In conclusion, our exploration into handwritten digit recognition epitomizes the intersection of machine learning, pattern recognition, and the practical world of handwritten documents. It accentuates the significance of accurate digit recognition in an increasingly digitized world, where automation and data-driven processes hold immense value. As technology continues to evolve, our project serves as a testament to the transformative potential of machine learning in advancing the accuracy and efficiency of handwritten digit recognition systems.

**ACKNOWLEDGENT**

The " Handwritten Digit Recognition" project has reached fruition thanks to the invaluable guidance and support of Professor Ashalesh Nayak. His mentorship and expertise in the field of computer vision have been instrumental in shaping the direction and success of this project.

Professor Ashalesh Nayak maam’s unwavering commitment to fostering innovation and research within the Department of Computer Science and Engineering at Manipal Institute of Technology, Manipal Academy of Higher Education, has provided the ideal environment for our project to thrive.

We extend our deepest appreciation for his insights, encouragement, and the invaluable knowledge he has imparted to our team throughout the course of this project. His mentorship has not only enhanced our technical skills but has also instilled a passion for pushing the boundaries of computer vision.

We would also like to thank the creators of the datasets and resources that underpinned our research, providing the critical foundation for our system's development.

In closing, we acknowledge Professor Ashalesh Nayak maam’s pivotal role in guiding this project and express our profound gratitude for his support and mentorship.

**REFERENCES**

1. LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

2. Ciresan, D., Meier, U., Masci, J., Gambardella, L. M., & Schmidhuber, J. (2012). Multi-column deep neural networks for image classification. Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

3. Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580.

4. Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep Learning (Vol. 1). MIT press Cambridge.

5. Simard, P. Y., Steinkraus, D., & Platt, J. C. (2003). Best practices for convolutional neural networks applied to visual document analysis. Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR).

6. Lecun, Y., Cortes, C., & Burges, C. (2010). MNIST handwritten digit database. AT&T Labs.