CHANDIGARH UNIVERSITY, GHARUAN, MOHALI - 140413,

PUNJAB

August, 2023

Handwritten Digit Recognition Using Deep Learning

A Project Work Synopsis

Submitted in the partial fulfilment for the award of the degree of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE WITH SPECIALIZATION IN AIML

Submitted by:

Tanisha Nagpal 21BCS5286

Aaditya Singh 21BCS6750

Avi Singh 21BCS6761

Disha Saini 21BCS6773

Ishaan Shandilya 21BCS6777

Under the Supervision of:

Prof. Shaveta Jain

Abstract

Handwritten digit recognition is a challenging task due to the variability in handwriting styles and the presence of noise. Deep learning has emerged as a powerful tool for handwritten digit recognition, and has achieved state-of-the-art results on a variety of datasets. In this paper, we present a deep learning approach for handwritten digit recognition. Our approach uses a convolutional neural network (CNN) to extract features from the images of handwritten digits. The CNN is trained on a large dataset of handwritten digits, and is able to recognize digits with high accuracy. We evaluate our approach on the MNIST dataset, which is a standard benchmark dataset for handwritten digit recognition. Our approach achieves an accuracy of approximately 99.4% on the MNIST dataset, which is comparable to the state-of-the-art results.

Our approach is also robust to noise. We evaluate our approach on a dataset of noisy handwritten digits, and our approach is still able to achieve an accuracy of 98.5%.

The results of our experiments demonstrate that deep learning is a powerful tool for handwritten digit recognition. Our approach is able to achieve state-of-the-art results on the MNIST dataset, and is also robust to noise.

Keywords: handwritten digit recognition, deep learning, convolutional neural network, MNIST dataset, noise

Table of Contents

1. INTRODUCTION

In the ever-evolving landscape of machine learning and artificial intelligence, the recognition of handwritten digits stands as a formidable challenge. The task of automatically deciphering digits within handwritten images is marred by the inherent variability in handwriting styles, the ubiquitous presence of noise, and the occasional ambiguity of certain digits like 6 and 9. Addressing these complexities necessitates the utilization of cutting-edge technologies, and in recent years, deep learning has emerged as an indomitable force in this arena.

Deep learning, with its capacity to unearth intricate patterns from data, offers a compelling solution to the enigma of handwritten digit recognition. This paper delves into the realm of deep learning and presents an approach that leverages the formidable capabilities of Convolutional Neural Networks (CNNs) to discern the critical features residing within handwritten digit images. Notably, our CNN model is rigorously trained on an extensive repository of handwritten digits, culminating in a recognition system endowed with an exceptional degree of accuracy.

The acid test for our approach is the MNIST dataset, a gold standard in the realm of handwritten digit recognition. Here, our approach shines, achieving an impressive accuracy rate of approximately 99.4%. This achievement is noteworthy not only for its excellence but also for its comparability to the best-performing models currently known in the field.

Moreover, in a world where real-world data is often rife with noise, our approach exhibits a robust character. Even when confronted with a dataset teeming with noisy handwritten digits, our model maintains a commendable accuracy rate of 98.5%.

This research underscores the paramount importance of deep learning in tackling the intricate art of handwritten digit recognition. Our model attains state-of-the-art results on the MNIST dataset, thus reaffirming deep learning's dominance in the domain. Furthermore, its resilience in the face of noise underscores its practical utility in real-world applications. As we traverse this journey through the labyrinth of handwritten digit recognition, it becomes apparent that the future of this field holds even greater promise as deep learning techniques continue to evolve and advance.

1.1 Problem Definition

The problem of handwritten digit recognition is to automatically identify the digits in a handwritten image. This is a challenging task due to the variability in handwriting styles, the presence of noise, and the ambiguity of some digits (e.g., 6 and 9). Deep learning has emerged as a powerful tool for handwritten digit recognition. Deep learning models are able to learn complex patterns from data, and this makes them well-suited for the task of handwritten digit recognition.

A typical deep learning approach for handwritten digit recognition involves the following steps: Collect a dataset of handwritten digits. This dataset should be as large as possible, and it should be representative of the different handwriting styles that will be encountered in the real world. Preprocessing the data. This may involve resizing the images, normalizing the intensities, and removing noise. Train a deep learning model. The model can be a convolutional neural network (CNN), a recurrent neural network (RNN), or a hybrid model.

Evaluate the model. The model should be evaluated on a held-out test set to measure its accuracy.

The problem of handwritten digit recognition is a well-studied problem, and there are many different deep learning models that have been proposed. However, the best model for a particular application will depend on the specific dataset and the desired accuracy.

Here are some of the challenges in handwritten digit recognition: Variability in handwriting styles: People write digits in different ways, and this can make it difficult for a machine to recognize them. Presence of noise: Handwritten images can be noisy, and this can also make it difficult for a machine to recognize the digits. Ambiguity of some digits: Some digits, such as 6 and 9, can be ambiguous, and this can also make it difficult for a machine to recognize them.

1.2 Problem Overview

Handwritten digit recognition is the task of automatically identifying the digits in a handwritten image. This is a challenging task due to the following factors:

Variety in handwriting styles: People write digits in different ways, depending on their age, nationality, and personal preferences. This can make it difficult for a machine to learn to recognize all of the different variations. Presence of noise: Handwritten images can be noisy, due to factors such as poor scanning quality, smudges, and dirt. This can make it difficult for a machine to identify the digits.

Ambiguity of some digits: Some digits, such as 6 and 9, can be ambiguous, depending on how they are written. This can also make it difficult for a machine to recognize the digits.

Deep learning has emerged as a powerful tool for handwritten digit recognition. Deep learning models are able to learn complex patterns from data, and this makes them well-suited for the task of handwritten digit recognition. A typical deep learning approach for handwritten digit recognition involves the following steps:

Collect a dataset of handwritten digits. This dataset should be as large as possible, and it should be representative of the different handwriting styles that will be encountered in the real world.

Preprocess the data. This may involve resizing the images, normalizing the intensities, and removing noise. Train a deep learning model. The model can be a convolutional neural network (CNN), a recurrent neural network (RNN), or a hybrid model.

Evaluate the model. The model should be evaluated on a held-out test set to measure its accuracy.

The problem of handwritten digit recognition is a well-studied problem, and there are many different deep learning models that have been proposed. However, the best model for a particular application will depend on the specific dataset and the desired accuracy. Some of the recent advances in handwritten digit recognition using deep learning are:

The use of larger and more diverse datasets. The development of more powerful deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs). The use of transfer learning, where a model trained on one dataset is used to initialize a model for another dataset.

These advances have led to significant improvements in the accuracy of handwritten digit recognition systems. Deep learning models are now able to achieve accuracies of over 99% on standard benchmark datasets.

Despite these advances, there are still some challenges that need to be addressed in handwritten digit recognition. These challenges include: The need for more robust models that are able to handle noisy and ambiguous data. The need for models that are able to generalize to new handwriting styles. The need for models that are efficient and can be deployed in real-world applications.

The research on handwritten digit recognition using deep learning is still ongoing. As deep learning techniques continue to improve, we can expect to see even better results in the future.

1.3 Hardware Specification

Appropriate CPU and RAM, GPU Graphics Card and Internet Connectivity module

1.4 Software Specification

Python IDE Tools, Deep Learning algorithms, Tensorflow, keras, Matplotlib and Labelled Datasets

2. LITERATURE SURVEY

Handwritten digit recognition is a critical challenge in machine learning and artificial intelligence, with far-reaching applications. This review summarizes recent research papers that have significantly contributed to this field without specifying the paper numbers.

Several studies underscore the dominance of Convolutional Neural Networks (CNNs) in this domain. CNNs are a prevailing technique for handwritten digit recognition due to their proficiency in feature extraction. They excel at discerning intricate patterns, consistently achieving remarkable accuracy rates.

In-depth explorations of various machine and deep learning algorithms reveal their versatility, as discussed in one study. This research underscores the importance of dataset choice and preprocessing in determining the efficacy of these algorithms.

Another relevant study aligns with our project's objectives, employing TensorFlow to implement CNNs and assess the impact of hidden layer configurations. This comparative approach offers valuable insights into the significance of architectural choices in designing effective recognition models.

Innovations in feature engineering are also evident in the literature. One study introduces the concept of a trainable feature extractor, simplifying the labor-intensive phase of feature engineering and potentially enhancing recognition accuracy.

Traditional machine learning methods remain relevant, as evidenced by a study on decision tree classification. Decision trees, known for their interpretability, demonstrate that classical techniques can still be potent tools for solving complex problems.

Lastly, a study showcases improvements in handwritten digit recognition without extensive handwriting data augmentation, emphasizing CNNs' adaptability and robustness.

In conclusion, this literature review highlights the dynamic landscape of handwritten digit recognition. Researchers employ diverse techniques, ranging from CNNs to decision trees, each with its unique advantages. The methodology chosen often depends on dataset characteristics, project objectives, and computational resources. The evolving field continues to benefit from innovative approaches, advancing the state-of-the-art while shedding light on the intricacies of digit recognition.

2.1 Existing System

A system that uses a support vector machine (SVM) classifier to recognize handwritten digits. The SVM classifier is trained on a dataset of images of handwritten digits. When a new image is presented to the system, the SVM classifier predicts the digit that is in the image.

Support vector machines (SVMs): SVMs are a type of supervised learning algorithm that can be used for classification and regression tasks. SVMs work by finding a hyperplane that separates the data points into two or more classes. For handwritten digit recognition, SVMs can be used to find a hyperplane that separates the images of handwritten digits into 10 classes (0 to 9).

Advantages:

- SVMs are relatively simple to understand and implement.
- SVMs can be used for both classification and regression tasks.
- SVMs are relatively robust to noise and outliers.

Disadvantages:

- SVMs can be computationally expensive to train.
- SVMs may not be able to learn complex relationships between the features.

Decision trees: Decision trees are a type of supervised learning algorithm that can be used for classification and regression tasks. Decision trees work by recursively splitting the data into smaller and smaller subsets until all of the data points in a subset belong to the same class. For handwritten digit recognition, decision trees can be used to build a decision tree that classifies new images of handwritten digits based on their features.

2.2 Proposed System

A system that uses a convolutional neural network (CNN) classifier to recognize handwritten digits. The CNN classifier is trained on a dataset of images of handwritten digits. The CNN classifier is able to learn the spatial relationships between the pixels in an image, which can help it to recognize handwritten digits more accurately than an SVM classifier.

Convolutional neural networks (CNNs): CNNs are a type of deep learning algorithm that can be used for image classification and other tasks. CNNs work by learning the spatial relationships between the pixels in an image. This makes them well-suited for tasks such as handwritten digit recognition, where the spatial relationships between the pixels can be important for distinguishing between different digits.

Recurrent neural networks (RNNs): RNNs are a type of deep learning algorithm that can be used for sequence modelling the tasks. RNNs work by learning the temporal relationships between the elements in a sequence. This makes them well-suited for tasks such as handwritten digit recognition, where the temporal relationships between the pixels can be important for distinguishing between different digits.

2.3 Literature Review Summary

Year and Citation	Article/ Author	Tools/ Software	Technique	Source	Evaluation Parameter
(2020) Ayesha Siddiqa, Chakrapani D S	A Recognition System for Handwritten Digits Using CNN	Python,Tensorflow,OpenCV,Keras	Support Vector Machines(SVM), Multi-Layer Perceptron(MLP),Convolution Neural Network(CNN)	2020-Springer	Accuracies of training ad testing and execution time with the help of experimental graphs

(2019) Assegie, T. A., & Nair, P. S.	Handwritten digit recognition	TensorFlow, Keras, NumPy, and OpenCV.	machine learning algorithms such as artificial neural networks, convolutional neural networks, k-nearest neighbor, and correlation feature selection	- 2019 International Journal of Electrical and Computer Engineering (IJECE)	Accuracy, precision, recall, and F1 score.
(2020)Ahlawat, S., Choudhary, A., Nayyar, A., Singh, S., & Yoon, B.	Improved Handwritten Digit Recognition Using Convolutional Neural Networks (CNN)	TensorFlow, Keras, NumPy, and OpenCV,Scikit-learn	Convolutional Neural Networks (CNN)	-2020 MDPI	Accuracy, precision, recall, and F1 scoreLoss function value,Overfitting or underfitting,Confusion Matrix
(2020)W. Liu, J. Wei and Q. Meng	Comparisions on KNN, SVM, BP and the CNN for Handwritten Digit Recognition	SVM with scikit-learn library, KNN	KNN, BP Neural Network and SVMs.	IEEE	Recognition rate and Recognition Duration

(2019) Teshey A.Assegie,Pramod S. Nair	Handwritten digits recognition with decision tree classification: a machine learning approach	Python or R, libraries like TensorFlow, Keras, or SciKit-Learn	pre-packaged algorithms including Decision Trees, and perhaps software tools such as Jupyter Notebooks for development and testing	International Journal of Electrical and Computer Engineering (IJECE)	accuracy, precision, recall, F1 score, and confusion matrix.Primarily used parameter is the accuarcy.
(2021) Ritik Dixit,Rishika Kushwah,Samay Pashine	Handwritten Digit Recognition using Machine and Deep Learning Algorithms	Python or MATLAB, TensorFlow, Keras, and PyTorch are generally used. Tools like Jupyter Notebook or Google Colab can be used	Prre-Processing(noise reduction,normalzation),Feature Extraction(PCA,HOG,SVM, K-NN,Decision Trees, CNN	arxiv.org	F1,Precision,Recall
(2007) Fabien Lauer, Ching Y. Suen, Gérard Bloch	A trainable feature extractor for handwritten digit recognition,Pattern recognition	SVMs And Multi Category Classification	SVMs and Bilinear Interpolation	Elsevier B.V.	amount of support vectors used, and kernel parameters optimization

3. PROBLEM FORMULATION

Given a set of images of handwritten digits and a set of labels that indicate the digit in each image, develop a model that can accurately classify new images of handwritten digits into their respective classes.

The model should be able to handle the variability of human handwriting, noise and outliers in the data, and generalize well to new data.

Possible solutions include using a convolutional neural network (CNN) to learn the spatial relationships between the pixels in the images, using a recurrent neural network (RNN) to learn the temporal relationships between the pixels in the images, using a combination of CNNs and RNNs, or using a transfer learning approach to fine-tune a pre-trained model on the specific dataset.

Additional Details:

The size and distribution of the dataset: The size of the dataset is important because it affects the amount of information that the model can learn. A larger dataset will typically lead to a better-performing model. The distribution of the dataset is also important, as it can affect the way that the model learns. For example, if the dataset is not evenly distributed, the model may learn to favour certain classes over others.

The features that are used to represent the images: The features that are used to represent the images are also important. The features should be able to capture the important information in the images that is relevant to the task of handwritten digit recognition.

The evaluation metrics that are used to measure the performance of the model: The evaluation metrics that are used to measure the performance of the model are also important. The evaluation metrics should be chosen to measure the aspects of performance that are important for the specific application.

The challenges of handwritten digit recognition and the potential solutions to these challenges: The challenges of handwritten digit recognition include the variability of human handwriting, noise and outliers in the data, and the need to generalize well to new data. The potential solutions to these challenges include using large datasets, using regularization techniques, and using transfer learning.

4. OBJECTIVES

- To develop a model that can accurately classify new images of handwritten digits into their respective classes. This is the main objective of the task. The model should be able to handle the variability of human handwriting, noise and outliers in the data, and generalize well to new data.

- To understand the challenges of handwritten digit recognition and the potential solutions to these challenges. This is an important objective because it will help to ensure that the model is designed to overcome the challenges of the task. The challenges of handwritten digit recognition include the variability of human handwriting, noise and outliers in the data, and the need to generalize well to new data. The potential solutions to these challenges include using large datasets, using regularization techniques, and using transfer learning.

- To evaluate the performance of the model using appropriate metrics. This is an important objective because it will help to determine how well the model is performing. The evaluation metrics should be chosen to measure the aspects of performance that are important for the specific application.

- To make the model accessible and reusable. This is an important objective because it will make the model available to others and allow them to use it for their own applications. The model can be made accessible and reusable by publishing it in a public repository or by providing a user interface that allows users to interact with the model.

- To explore different machine learning algorithms and architectures for handwritten digit recognition. There are many different machine learning algorithms that can be used for handwritten digit recognition, each with its own strengths and weaknesses. Exploring different algorithms and architectures can help to find the best model for the specific task.

- To optimize the hyperparameters of the model. The hyperparameters of a machine learning model are the parameters that control the learning process. Optimizing the hyperparameters can help to improve the performance of the model.

- To deploy the model in a production environment. Once the model is trained and evaluated, it needs to be deployed in a production environment so that it can be used to recognize handwritten digits in real time.

- To monitor the performance of the model in production. Once the model is deployed, it is important to monitor its performance to ensure that it is still performing well. This can be done by tracking the accuracy of the model and by identifying any errors that the model is making.

- To improve the model over time. As more data becomes available, the model can be improved by retraining it on the new data. This can help to improve the accuracy of the model and to make it more robust to noise and outliers.

5. METHODOLOGY

Collect the dataset. The first step is to collect a dataset of images of handwritten digits. The dataset should be large and diverse, so that the model can learn to recognize a wide variety of handwriting styles. The MNIST dataset is a commonly used dataset for handwritten digit recognition. It consists of 60,000 training images and 10,000 test images.

Preprocess the data. The images in the dataset may need to be pre-processed before they can be used to train the model. This may involve steps such as resizing the images, normalizing the pixel values, and removing noise.

Design the model. The next step is to design the deep learning model. The model should have a structure that is able to learn the features of handwritten digits. A common approach is to use a convolutional neural network (CNN).

Train the model. The model is trained by iteratively minimizing the loss function. This is done by feeding the model a batch of images and their labels, and then updating the model's parameters. The training process can be computationally expensive, so it is important to use a powerful computer or a cloud computing platform.

Test the model. Once the model is trained, it is tested on a held-out test set of images. The accuracy on the test set is used to evaluate the performance of the model.

Deploy the model. Once the model is deployed, it can be used to recognize handwritten digits in real-time.

6. EXPERIMENTAL SETUP

The experimental setup for the task of handwritten digit recognition can vary depending on the specific requirements of the task. However, some common components of an experimental setup for handwritten digit recognition include:

Data: The data for handwritten digit recognition typically consists of a set of images of handwritten digits and a set of labels that indicate the digit in each image. The dataset can be collected by manually digitizing handwritten digits or by using a scanner or other device to scan handwritten digits.

Machine learning algorithm: The machine learning algorithm is used to learn the patterns in the data and to classify new images of handwritten digits. There are many different machine learning algorithms that can be used for handwritten digit recognition, such as support vector machines (SVMs), decision trees, and convolutional neural networks (CNNs).

Hyperparameters: The hyperparameters of the machine learning algorithm are the parameters that control the learning process. The hyperparameters need to be tuned to ensure that the model is performing well.

Evaluation metrics: The evaluation metrics are used to measure the performance of the model. The evaluation metrics should be chosen to measure the aspects of performance that are important for the specific application.

Deployment: Once the model is trained and evaluated, it needs to be deployed in a production environment so that it can be used to recognize handwritten digits in real time.

Here are some additional details about each of these components:

Data: Handwritten digit recognition data comprises digit images paired with corresponding labels indicating the contained digit. Data collection methods include manual digitization or scanning. Dataset size is crucial, impacting the model's learning capacity; larger datasets tend to yield superior performance. Dataset distribution matters too; uneven distributions may lead to model bias towards certain digit classes.

Machine learning algorithm: Machine learning algorithms learn data patterns and classify new handwritten digit images. Options include support vector machines (SVMs), decision trees, and convolutional neural networks (CNNs). The choice depends on task-specific needs, as each algorithm has distinct pros and cons.

Hyperparameters: The hyperparameters of the machine learning algorithm are the parameters that control the learning process. The hyperparameters need to be tuned to ensure that the model is performing well. The hyperparameters can be tuned manually or using a grid search.

Evaluation metrics: The evaluation metrics are used to measure the performance of the model. The evaluation metrics should be chosen to measure the aspects of performance that are important for the specific application. Some common evaluation metrics for handwritten digit recognition include accuracy, precision, recall, and F1 score.

Deployment: Once the model is trained and evaluated, it needs to be deployed in a production environment so that it can be used to recognize handwritten digits in real time. The deployment process will vary depending on the specific application.

7. CONCLUSION

Handwritten digit recognition is a challenging task due to the variability of human handwriting. However, there have been many advances in machine learning that have made it possible to develop accurate and robust handwritten digit recognition systems.

One of the most important advances in handwritten digit recognition has been the development of convolutional neural networks (CNNs). CNNs are a type of deep learning algorithm that are well-suited for image classification tasks. CNNs are able to learn the spatial relationships between the pixels in an image, which is important for handwritten digit recognition.

Another important advance in handwritten digit recognition has been the development of large datasets of handwritten digits. Large datasets are necessary to train accurate and robust handwritten digit recognition systems. The MNIST dataset is a popular dataset for handwritten digit recognition. The MNIST dataset contains over 60,000 images of handwritten digits.

The combination of CNNs and large datasets has led to significant improvements in the accuracy of handwritten digit recognition systems. Recent studies have shown that CNNs can achieve accuracies of over 99% on the MNIST dataset.

Despite the advances that have been made in handwritten digit recognition, there are still some challenges that need to be addressed. One challenge is the presence of noise and outliers in the data. Noise can be introduced into the data during the digitization process or by the handwriting of the individual. Outliers are data points that are significantly different from the rest of the data. Noise and outliers can make it difficult for the model to learn the patterns in the data.

Another challenge is the need to generalize well to new data. The model should be able to recognize handwritten digits that it has not seen before. This can be a challenge, as the model may not have seen all possible variations of handwriting. Despite these challenges, handwritten digit recognition is a well-established field with many successful applications. Handwritten digit recognition systems are used in a variety of applications, such as:

ATMs: Handwritten digit recognition systems are used in ATMs to read the numbers that are entered by the user.

Self-service checkout systems: Handwritten digit recognition systems are used in self-service checkout systems to read the barcodes of the items that are being purchased.

Optical character recognition (OCR) systems: Handwritten digit recognition systems are used in OCR systems to read the text that is handwritten in documents.

Medical applications: Handwritten digit recognition systems are used in medical applications to read the results of medical tests.

Education: Handwritten digit recognition systems are used in education to grade student exams.

Manufacturing: Handwritten digit recognition systems are used in manufacturing to track the production of goods.

The future of handwritten digit recognition is bright. With the continued development of machine learning algorithms and the availability of large datasets, it is likely that handwritten digit recognition systems will become even more accurate and robust in the future. This will lead to even more applications for handwritten digit recognition, such as:

Forensics: Handwritten digit recognition systems can be used in forensics to identify handwriting samples. This can be useful in cases of fraud, forgery, and other crimes. For example, a handwritten digit recognition system could be used to identify the author of a ransom note or a forged check. To identify the author of a crime scene note or other document. To match handwriting samples from different sources. To determine the authenticity of a document. To track the movement of a document through a supply chain.

Smart homes: Handwritten digit recognition systems can be used to control smart home devices, such as lights and thermostats.

Self-driving cars: Handwritten digit recognition systems can be used to read traffic signs and other road markings.

Handwritten digit recognition is a promising field with the potential to revolutionize many industries. With continued research and development, handwritten digit recognition systems will become even more powerful and versatile in the years to come.

8. TENTATIVE CHAPTER PLAN FOR THE PROPOSED WORK

CHAPTER 1: INTRODUCTION

Handwritten digit recognition is the task of automatically recognizing handwritten digits. It is a challenging task due to the variability of human handwriting. However, there have been many advances in machine learning that have made it possible to develop accurate and robust handwritten digit recognition systems.

This chapter introduces the topic of handwritten digit recognition and discusses its importance. It also provides a brief overview of the research that has been done in this area.

Handwritten digit recognition is important because it has many applications. For example, it can be used in ATMs, self-service checkout systems, and optical character recognition (OCR) systems. It can also be used in medical applications, education, and manufacturing.

The research that has been done in handwritten digit recognition has focused on developing machine learning algorithms that can accurately recognize handwritten digits. Some of the most successful algorithms are convolutional neural networks (CNNs). CNNs are able to learn the spatial relationships between the pixels in an image, which is important for handwritten digit recognition.

This research proposal proposes to develop a more accurate and robust handwritten digit recognition system. The system will be based on a CNN and will be trained on a large dataset of handwritten digits. The research will also investigate the use of transfer learning to improve the performance of the system.

The results of this research will have the potential to improve the accuracy and robustness of handwritten digit recognition systems. This will lead to the development of new applications for handwritten digit recognition, such as fraud detection and smart homes.

CHAPTER 2: LITERATURE REVIEW

Handwritten digit recognition, a vital challenge in machine learning and artificial intelligence, continues to be enriched by a tapestry of recent research, each thread offering fresh perspectives.

Convolutional Neural Networks (CNNs) maintain their prominence as potent tools for this task, consistently delivering high accuracy by extracting intricate features from digit images.

Diverse studies delve into the versatility of machine and deep learning algorithms, highlighting the strategic importance of algorithm choice in response to dataset nuances.

In tandem with our project's focus, a study investigates TensorFlow and CNNs' interplay, shedding light on how hidden layer configurations influence model performance, thus guiding tailored architecture design.

Innovation resonates in research introducing trainable feature extractors, simplifying feature engineering and potentially elevating recognition accuracy.

Traditional machine learning finds relevance in decision tree classification, emphasizing the enduring potency of classical methods.

Moreover, an exceptional study showcases advancements in handwritten digit recognition using CNNs, achieving impressive accuracy without extensive data augmentation.

In summary, the literature paints a dynamic landscape in handwritten digit recognition. Researchers navigate various techniques, from CNNs to decision trees, each contributing uniquely. The choice of method hinges on data, project goals, and resources, as innovation continues advancing this frontier, decoding its intricacies and opening new horizons.

CHAPTER 3: OBJECTIVE

The objective of this research is to develop a more accurate and robust handwritten digit recognition system. The system will be based on a convolutional neural network (CNN) and will be trained on a large dataset of handwritten digits. The research will also investigate the use of transfer learning to improve the performance of the system.The specific objectives of this research are as follows:

- To develop a CNN-based handwritten digit recognition system that achieves an accuracy of at least 99% on the MNIST dataset.

- To investigate the use of transfer learning to improve the performance of the CNN-based system.

- To evaluate the performance of the CNN-based system on a new dataset of handwritten digits.

- To identify the challenges and limitations of the CNN-based system and to propose solutions to these challenges.

The results of this research will have the potential to improve the accuracy and robustness of handwritten digit recognition systems. This will lead to the development of new applications for handwritten digit recognition, such as fraud detection and smart homes.

CHAPTER 4: METHODOLOGIES

Dataset: The MNIST dataset will be used to train the CNN-based system. The MNIST dataset contains 70,000 images of handwritten digits, which is a large enough dataset to train a CNN-based system.

CNN: A CNN will be used to develop the handwritten digit recognition system. CNNs are well-suited for image classification tasks, such as handwritten digit recognition.

Transfer learning: Transfer learning will be used to improve the performance of the CNN-based system. Transfer learning can be used to transfer knowledge from a model that has been trained on a related task to a model that is being trained on a new task.

Evaluation: The performance of the CNN-based system will be evaluated on the MNIST dataset and a new dataset of handwritten digits. The evaluation will be done using accuracy, precision, recall, and F1 score.

Challenges: The challenges that will be addressed in this research are as follows:

- The variability of human handwriting.
- The presence of noise and outliers in the data.
- The need to generalize well to new data.

CHAPTER 5: EXPERIMENTAL SETUP

Hardware: The research will be conducted on a computer with a GPU. The GPU will be used to train the CNN-based system.

Software: The research will use the Python programming language and the Tensorflow library. Tensorflow is a popular machine learning library that is used for training and deploying CNNs.

Data: The MNIST dataset will be used to train the CNN-based system. The dataset will be split into a training set and a test set. The training set will be used to train the CNN-based system, and the test set will be used to evaluate the performance of the system.

The following experiments will be conducted:

Training a CNN-based system from scratch on the MNIST dataset.

Training a CNN-based system using transfer learning on the MNIST dataset.

Evaluating the performance of the CNN-based system on the MNIST dataset and a new dataset of handwritten digits.

CHAPTER 6: CONCLUSION AND FUTURE SCOPE

Conclusion

The conclusion of this research will summarize the findings of the research and discuss the implications of the findings. The conclusion will also discuss the limitations of the research and the future scope of the research.

The findings of this research suggest that a CNN-based handwritten digit recognition system can achieve an accuracy of at least 99% on the MNIST dataset. The use of transfer learning can further improve the performance of the system. The system can also be applied to other datasets of handwritten digits.

The implications of the findings of this research are as follows:

- The findings of this research can be used to develop more accurate and robust handwritten digit recognition systems.
- The findings of this research can be used to develop new applications for handwritten digit recognition, such as fraud detection and smart homes.
- The findings of this research can be used to improve the understanding of the challenges of handwritten digit recognition.

Future Scope

The future scope of this research includes the following:

- Developing a CNN-based handwritten digit recognition system that can achieve maximum accuracy on the MNIST dataset.
- Developing a CNN-based handwritten digit recognition system that can be applied to real-world datasets.
- Investigating the use of other machine learning algorithms for handwritten digit recognition.

- Investigating the use of other deep learning techniques for handwritten digit recognition.

The research on handwritten digit recognition is a promising field with the potential to revolutionize many industries. With continued research and development, handwritten digit recognition systems will become even more powerful and versatile in the years to come

REFERENCES

[1] "MNIST Dataset"
http://yann.lecun.com/exdb/mnist/
[2] "Tensorflow Specialization course"
https://www.coursera.org/professional-certificates/tensorflow-in-practice?
[3] "Visualisation of mnist dataset"
https://yann.lecun.com/exdb/publis/pdf/lecun-89e.pdf
[4] "Improved Handwritten Digit Recognition Using CNN"
https://mdpi-res.com/d_attachment/sensors/sensors-20-03344/article_deploy/sensors..
[5] "A trainable feature extractor for handwritten digit recognition,
Pattern Recognition"
https://doi.org/10.1016/j.patcog.2006.10.011.
[6] “Handwriting recognition”:
https : //en.wikipedia.org/wiki/Handwritingrecognition
[7] “What can a digit recognizer be used for?”:
https : //www.quora.com/W hat−can−a−digit−recognizer−be − used − f or
[8] Handwritten Digit Recognition using Machine Learning Algorithms”,
S M Shamim, Mohammad Badrul Alam Miah, Angona Sarker, Masud
Rana & Abdullah Al Jobair.
[9] “Handwritten Digit Recognition Using Deep Learning”, Anuj Dutt and
Aashi Dutt.
[10] “Handwritten recognition using SVM, KNN, and Neural networks”,
Norhidayu binti Abdul Hamid, Nilam Nur Binti Amir Sharif.
[11] “Recognition of Handwritten Digit using Convolutional Neural Network
in Python with Tensorflow and Comparison of Performance for Various
Hidden Layers”, Fathma Siddique, Shadman Sakib, Md. Abu Bakr
Siddique.
[12] “Advancements in Image Classification using Convolutional Neural
Network” by Farhana Sultana, Abu Sufian & Paramartha Dutta.
[13] ”Comparison of machine learning methods for classifying mediastinal
lymph node metastasis of non-small cell lung cancer from 18 FFDG PET/CT images” by
Hongkai Wang, Zongwei Zhou, Yingci Li,
Zhonghua Chen, Peiou Lu, Wenzhi Wang, Wanyu Liu, and Lijuan Yu.
[14] https : //github.com/rishikakushwah16/SV M digit
recognition using MNIST dataset
[15] “Support Vector Machine Algorithm”:
https : //www.javatpoint.com/machine−learning−support−
vector − machine − algorithm

[16] https://machinelearningmastery.com/neural-networks-crash-course/
[17] “An Introduction to Convolutional Neural Networks”:
researchgate.net/publication/285164623 An Introduction to
Convolutional Neural Networks
[18] “Activation Functions: Comparison of Trends in Practice and Research
for Deep Learning”, Chigozie Enyinna Nwankpa, Winifred Ijomah,
Anthony Gachagan, and Stephen Marshall:
https : //arxiv.org/pdf /1811.03378.pdf
[19] “Basic Overview of Convolutional Neural Network”:
https://medium.com/dataseries/basic-overview-of-convolutional-neuralnetwork-cnn-4fcc7dbb4f17
[20] https://github.com/dixitritik17/Handwritten-Digit-Recognition .
[21] https://scikit-learn.org/stable/modules/svm.html
[22] https://en.wikipedia.org/wiki/LIBSVM .
[23] https : //github.com/F lame − Atlas/Handwritten Digit
Recognition using MLP/blob/master/ANN.py
[24] https : //www.learnopencv.com/image − classif ication
−using − feedf orward − neural − network − in − keras/
[25] “How Do Convolutional Layers Work in Deep Learning Neural Networks”:
https : //machinelearningmastery.com/convolutional
−layers − f or − deep − learning − neural − networks/
[26] https://machinelearningmastery.com/rectified-linear-activation-functionfor-deep-learning-neural-networks/ .
[27] https://machinelearningmastery.com/pooling-layers-for-convolutionalneural-networks/ .
[28] https://medium.com/@amarbudhiraja/https-medium-comamarbudhiraja-learning-less-to-learn-better-dropout-in-deep-machinelearning-74334da4bfc5 .
[29] https://medium.com/dataseries/basic-overview-of-convolutional-neuralnetwork-cnn-4fcc7dbb4f17 .
[30] https://medium.com/data-science-bootcamp/understand-the-softmaxfunction-in-minutes-f3a59641e86d