World Journal of Advanced Engineering Technology and Sciences

eISSN: 2582-8266

Cross Ref DOI: 10.30574/wjaets

Journal homepage: https://wjaets.com/

WJAETS World Journal of Advanced Engineering Technology and Sciences

(Research Article)

Implementing handwritten text recognition using deep learning with TensorFlow: An MNIST dataset approach

Ogochukwu Patience Okechukwu [1, *], Godson Nnaeto Okechukwu [2], Okwuchukwu Ejike Chukwuogo [1] and Amanda Uloma Anyigor-Ogah [1]

[1] *Department of Computer Science, Nnamdi Azikiwe University, Awka, Nigeria.*

[2] *Department of Electronic/Computer Engineering, Nnamdi Azikiwe University, Awka, Nigeria.*

World Journal of Advanced Engineering Technology and Sciences, 2024, 12(02), 544–552

Publication history: Received on 17 June 2024; revised on 29 July 2024; accepted on 01 August 2024

Article DOI: https://doi.org/10.30574/wjaets.2024.12.2.0320

Abstract

Handwritten text recognition (HTR) is a pivotal technology with extensive applications in document digitization, postal automation, and educational tools. This paper delves into the implementing a deep learning-based system for recognizing handwritten digits using TensorFlow and the MNIST dataset. The MNIST dataset, a widely-used benchmark, comprises 60,000 training images and 10,000 testing images of handwritten digits, each standardized to a 28x28 pixel grayscale format. Leveraging the power of Convolutional Neural Networks (CNNs), our model effectively extracts features and classifies digits with high accuracy. The model architecture consists of multiple convolutional layers for feature extraction, pooling layers for dimensionality reduction, and fully connected layers for classification. Preprocessing steps include normalizing pixel values and one-hot encoding the labels, ensuring the data is optimally formatted for training. The TensorFlow framework, known for its robustness and scalability, facilitates the development and deployment of the model. Through a series of experiments, the model demonstrates impressive performance, achieving high accuracy on the MNIST test set. This paper underscores the potential of deep learning in handwritten text recognition. It sets the stage for future enhancements, such as recognizing more complex handwritten texts and optimizing the system for practical applications. The results highlight the effectiveness of deep learning techniques in overcoming the challenges associated with handwritten text recognition, paving the way for advanced, real-world implementations.

Keywords: Handwritten Text Recognition (HTR); Deep Learning; TensorFlow; Convolutional Neural Network (CNN); Dataset; Image Processing; Machine Learning

1. Introduction

Handwritten text recognition (HTR) represents a cornerstone technology with wide-ranging applications in various fields such as document digitization, postal automation, banking, and educational tools. The ability to accurately interpret and digitize handwritten text enables more efficient data entry, archival, and retrieval processes, leading to significant improvements in productivity and accessibility. This technology is crucial for converting handwritten historical documents into digital formats, automating the sorting of mail and checks, and assisting in the real-time transcription of notes and forms.

Traditionally, HTR systems relied on handcrafted features and classical machine learning techniques, such as k-nearest neighbors, support vector machines, and decision trees. However, these methods often struggled with the inherent variability and complexity of human handwriting. Variations in writing styles, slant, pressure, and the connection

* Corresponding author: Ogochukwu Patience Okechukwu

between characters present significant challenges. Additionally, noise and distortions in scanned images can further complicate recognition tasks.

The advent of deep learning has revolutionized the field, providing powerful tools that can automatically learn to extract meaningful features from raw data and achieve unprecedented accuracy in recognition tasks. Deep learning models, particularly Convolutional Neural Networks (CNNs), have demonstrated superior performance in various image-processing tasks due to their ability to capture spatial hierarchies in images through convolutional layers. These layers perform localized filtering operations, allowing the network to detect and learn various features such as edges, textures, and shapes.

This paper focuses on developing a deep learning-based HTR system using TensorFlow, a leading open-source machine learning framework developed by Google. TensorFlow provides a comprehensive ecosystem for building and deploying machine learning models, with robust support for deep learning architectures [1]. The system is trained on the Modified National Institute of Standards and Technology (MNIST) dataset, which is widely regarded as the benchmark for handwritten digit recognition tasks. The MNIST dataset comprises 70,000 images of handwritten digits, split into 60,000 training images and 10,000 testing images. Each image is a 28x28 pixel grayscale image, representing digits from 0 to 9. Despite its simplicity, the MNIST dataset provides a foundational step for understanding more complex HTR tasks. The primary architecture utilized in this project is the Convolutional Neural Network (CNN), which is specifically designed to process and analyze visual data. CNN has demonstrated superior performance in a range of image recognition tasks due to its ability to capture spatial hierarchies in images through convolutional layers. These layers perform localized filtering operations, allowing the network to detect and learn various features such as edges, textures, and shapes.

The implementation process involves several critical steps, beginning with data preprocessing. This step ensures that the pixel values of the images are normalized to a range between 0 and 1, which helps in accelerating the training process and improving the model's performance. The labels are also converted to one-hot encoded vectors, a format that is compatible with the categorical nature of the classification task. The core of the project involves designing, training, and evaluating the CNN model. The model consists of multiple convolutional layers interspersed with pooling layers to progressively reduce the spatial dimensions of the feature maps. This reduction helps in making the computation more manageable and focusing the model on the most salient features. The final layers are fully connected and use the softmax activation function to output the probability distribution over the 10-digit classes.

Once the model is trained, its performance is evaluated on the test dataset to measure its accuracy and generalization capability. The high accuracy achieved by the model on the test set demonstrates the effectiveness of deep learning approaches in HTR tasks. This paper aims to provide a comprehensive overview of the application of deep learning in handwritten text recognition, showcasing the implementation details and the results achieved. By leveraging TensorFlow and the MNIST dataset, the paper also highlights the potential of CNNs in overcoming the challenges associated with recognizing handwritten digits and paving the way for future advancements in the field.

The paper is organized as follows: Section 2 presents an overview of related works. Subsequently, in section 3, the methodology is discussed. Section 4 shows the Application of Handwritten Text recognition. Finally, in section 5 conclusion and future direction are presented.

2. Related Works

HTR plays a critical role in bridging the gap between the physical and digital worlds. By deciphering handwritten characters, HTR unlocks a treasure trove of information trapped in handwritten documents, notes, and forms. This technology has vast applications, including automatic mail sorting, medical record processing, and historical document analysis. Building an accurate and efficient HTR system requires careful consideration of existing techniques and advancements in the field. This review of related works delves into the current landscape of HTR, focusing on approaches that leverage deep learning, particularly the TensorFlow framework. We will explore various methods used for preprocessing handwritten images, feature extraction, and character recognition, all to inform the development of the HTR system using TensorFlow and the MNIST dataset.

[2] tackles the complexities of automatically segmenting and recognizing unconstrained handwritten text, with a particular focus on the challenges presented by Persian script. The research acknowledges the inherent difficulties in dealing with unconstrained handwritten documents, including inconsistencies in character shapes and separations, variations in writing styles, and the presence of skewed or overlapping text lines. These challenges are further amplified when dealing with Persian handwriting, which has a distinct character set, writing order, and inherent cursives

compared to Latin-based scripts. The limitations of existing segmentation techniques designed for Latin scripts are highlighted. The paper further argues that these methods are not effective for the accurate segmentation of Persian handwritten documents due to the script's unique characteristics. To address these issues, the paper proposes a novel concept called "piece-wise painting" for text line segmentation. This method involves splitting the document into stripes, applying smoothing operations, binarization, and further smoothing to create a black and white representation, essentially transforming the document into a series of "painted" bands. Building upon this concept, two new text-line segmentation techniques specifically designed for Persian handwritten documents are introduced.

For documents with significant skew, a separate painting-based algorithm is developed to estimate and correct the skew at the page level to facilitate the segmentation process. However, this approach is not suitable for unconstrained documents with wavy, curved, or multi-skewed text lines. To handle these complexities, a baseline tracing/aligning method is introduced. This method leverages the characteristic of Persian text where characters within a word or sub-word are connected along a horizontal line called the baseline. Due to the cursive and overlapping nature of Persian handwriting, the work proposes bypassing word extraction and performing character segmentation directly. A character segmentation algorithm is developed that utilizes the baseline information to isolate individual characters.

Finally, to achieve text recognition and demonstrate the effectiveness of the proposed segmentation methods, recognition approaches for Persian handwritten characters and numerals are explored. These approaches utilize feature extraction techniques and classification methods like Nearest Neighbor and Support Vector Machines (SVMs). Recognizing the scarcity of comprehensive datasets for evaluating handwritten text recognition methods, the project undertook the creation of a new database containing handwritten text in various languages written by individuals from different age groups. Additionally, a specific database of segmented Persian handwritten characters was created to facilitate further research. The performance of the proposed algorithms is evaluated on existing and newly created datasets, paving the way for improved automatic segmentation and recognition of unconstrained Persian handwritten text.

[3] in their paper shows that extracting information from handwritten mathematical documents presents a unique challenge, unlike standardized text. The paper noted that mathematical notation relies on a vast repertoire of symbols and notations that can vary in complexity and individuality depending on the writer. The paper also addresses this challenge by proposing a system that accomplishes automatic recognition of handwritten mathematical text through a breakdown of the process into several crucial stages. The first hurdle involves **preprocessing** the handwritten document. This initial step likely involves tasks like noise reduction to eliminate smudges or imperfections from the scanned image. Additionally, skew correction might be necessary to account for documents that are not perfectly aligned. Finally, binarization, which converts the image to a clear distinction between foreground (ink) and background (page), is often a critical step in preparing the image for further processing. The paper maintained that once the image is prepped, the system tackles **text line segmentation**. Here, the focus shifts to separating the document into individual lines of text. This might involve identifying line breaks or following the writing pattern to isolate each line. Notably, these lines could include a mix of standard text and handwritten mathematical expressions, making segmentation even more critical. Having separated the lines, the system delves into **mathematical symbol segmentation**. This stage is likely the most intricate, as it requires pinpointing and isolating the individual mathematical symbols within each text line. The complexity arises from the inherent variability in how people write these symbols and the potential for overlapping characters that can be difficult to disentangle. Finally, with the symbols isolated, the system proceeds to **symbol recognition**. This involves the crucial task of assigning the correct mathematical meaning to each symbol. The system likely achieves this by employing a database of known mathematical symbols and matching the isolated symbol against entries in the database.

The paper [4], deals in the domain of pattern recognition with emphasis on HTR. HTR focuses on deciphering and interpreting handwritten text, encountered in various forms like scanned documents, touch screen input, or photographs. This research proposes a novel approach to tackle offline HTR using deep neural networks. The ever-growing availability of vast datasets and advancements in algorithms have made training deep neural networks significantly more accessible. Their system hinges on image segmentation, a technique that isolates individual characters within the handwritten text. By accurately identifying and segmenting characters, the proposed approach aims to achieve improved accuracy and efficiency in HTR across diverse applications and contexts.

X-ray weld seam images are a treasure trove of information for professionals in the field of welding; [5] believe that by leveraging the power of graphic-text recognition technology, these images can be harnessed to automate data collection in complex industrial environments, leading to significant improvements in work efficiency. The research delves into the application of deep learning to enhance the accuracy and efficiency of extracting weld seam information from X-ray images. The research commenced with the crucial step of amassing a comprehensive dataset of X-ray weld seam images specifically for training and evaluating the proposed model. The study tackles also the challenge in two key stages: text detection and text recognition. For **text detection**, the authors implemented a model based on the DBNet algorithm, a deep learning architecture adept at object detection. Furthermore, the research incorporates post-processing techniques specifically tailored to the unique characteristics of weld seam images. Through rigorous model training, the authors achieved an impressive performance in detecting text regions within the images. The model achieved a precision of 91%, a recall of 92.4%, and a combined F1 score of 91.7% on the test dataset. These metrics indicate the model's effectiveness in accurately pinpointing text locations within the X-ray images. The second stage focuses on **text recognition**. Here, the authors introduced modules like Channel Attention (CA), Convolutional Block Attention Module (CBAM), and Hierarchical Feature Aggregation (HFA) within the deep learning architecture. These modules play a critical role in effectively capturing both the character position information and the global features of the text lines within the images. These optimizations culminated in a remarkable text line recognition accuracy of 93.4%.

Managing electrical equipment efficiently hinges on readily accessing the vital information displayed on their nameplates. This research by [6] proposes a method for automatically extracting text information from these nameplates, leveraging the power of deep learning. Optical Character Recognition (OCR), a key component of Artificial Intelligence (AI), is further enhanced by deep learning to achieve superior accuracy in nameplate recognition and broaden its applicability. The OCR process focuses on two crucial aspects: text region detection and text recognition within those regions. The authors propose a deep learning model based on the concept of Meaningful Learning for OCR recognition. This model tackles text detection using the Connectionist Text Proposal Network (CTPN) algorithm, adept at pinpointing text regions within images. Once the text regions are identified, the model employs the Convolutional Recurrent Neural Network (CRNN) algorithm for text recognition within those regions. CRNNs excel at deciphering sequential data like text, making them well-suited for this task. By combining these deep learning techniques, the paper paves the way for automating text extraction from electrical nameplates. This approach has the potential to significantly improve equipment management efficiency by enabling the rapid and accurate retrieval of crucial information from nameplates.

Text recognition and detection play vital roles in various applications, and man has made much progress toward such applications as observed by [7], though many challenges remain. Such challenges include ambiguity of such text or its distortion, hindering accurate recognition. This highlights the need for continuous advancements in text recognition and detection algorithms. The research uses Artificial Intelligence (AI) technology to enhance these algorithms. The study explores the use of AI through experimentation and the result shows an improvement of at least 11% and up to 19% in accuracy. By use of AI, the development of more robust and accurate methods for text recognition and detection can be achieved and has the potential to revolutionize various fields that rely heavily on the ability to extract information from textual data.

The review of related works provides valuable insights into the current state of the art in handwritten text recognition. Here are some key takeaways that will guide the development of our HTR system using TensorFlow and the MNIST dataset:

The dominance of deep learning, particularly Convolutional Neural Networks (CNNs), in achieving high accuracy for HTR is evident. Our system will incorporate CNN architectures suitable for image recognition tasks.

Effective preprocessing techniques like normalization, noise reduction, and binarization play a crucial role in preparing the handwritten image data for further processing. We will integrate these techniques into our preprocessing pipeline.

3. Methodology

3.1. Data Acquisition and Preprocessing

The MNIST dataset is easily accessible through the TensorFlow library. It contains 60,000 training images and 10,000 test images of handwritten digits (0-9).

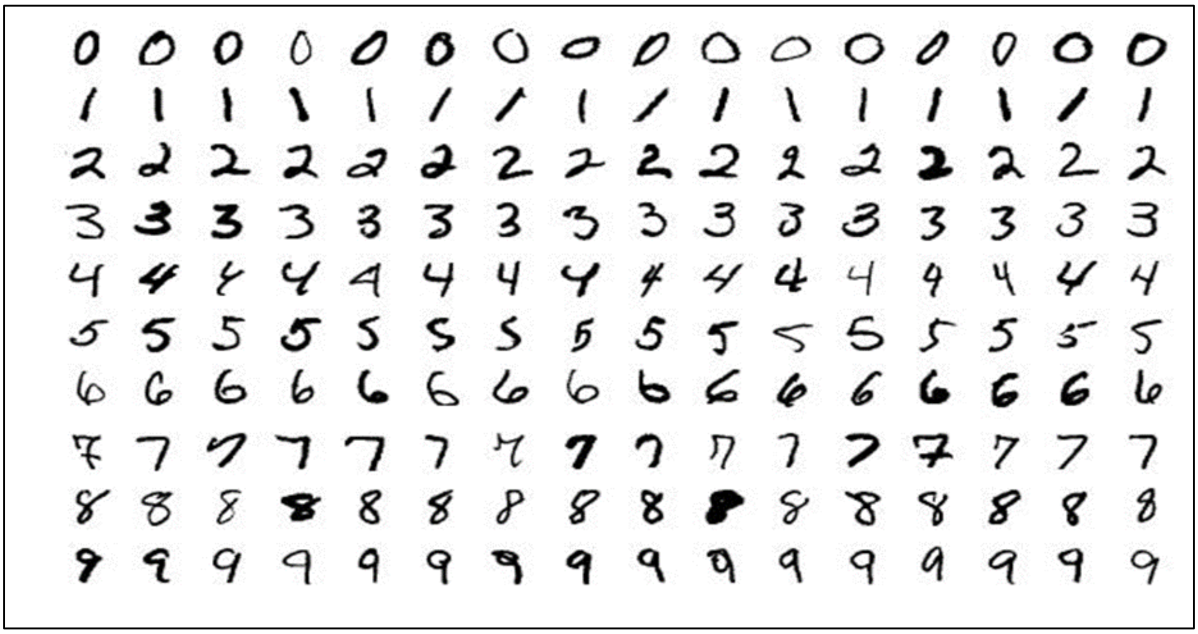

Figure 1 An overview of what the MNIST dataset looks like

3.2. Preprocessing the Data

Preprocessing the images is crucial to improve the model's performance. Common preprocessing steps include normalization, noise reduction, and binarization. For simplicity, the initial preprocessing step normalizes the pixel values to a range of [0, 1].

3.3. Data Augmentation

Data augmentation plays a crucial role in enhancing the performance of machine learning models, especially in the domain of image recognition. By applying random transformations to the existing images in the dataset, data augmentation helps create a more diverse training set. This increased diversity allows the model to generalize better, making it more robust and improving its performance on unseen data.

In our project, we used TensorFlow's ImageDataGenerator to implement data augmentation. Below are the specific techniques applied and their implementation:

- **Rotation**: We rotated images randomly within a range of 10 degrees to simulate different writing angles.
- **Width and Height Shifts**: We translated images horizontally and vertically within a range of 10% to mimic different positioning of the text.
- **Zoom**: We zoomed in and out on images within a range of 10% to create variations in text size.
- **Shear**: We applied shearing transformations within a range of 20% to simulate slanted handwriting.
- **Brightness Adjustment:** We modified the brightness of images to account for different lighting conditions, within a range of 80% to 120% of the original brightness.

By combining these transformations, we created a more robust and comprehensive dataset for training.

3.4. Building and Training the Model

A simple neural network model is built using TensorFlow's Keras API. The model consists of a flattening layer, a hidden dense layer with ReLU activation, and an output dense layer with softmax activation.

3.4.1. Training the Model

Train the model using the augmented data for better generalization.

3.4.2. Evaluating the Model

Evaluate the trained model on the test data to determine its accuracy.

3.4.3. Making Predictions

Use the trained model to make predictions on the test data and visualize the results.

The neural network model was trained using the augmented training data for four epochs. The accuracy achieved by the model at the end of each epoch was as follows:

```
ata from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
0434 [==============================] - 0s 0us/step

===========================] - 12s 5ms/step - loss: 0.2580 - accuracy: 0.9261 - val_loss: 16.9681 - val_accuracy: 0.9606

===========================] - 13s 7ms/step - loss: 0.1127 - accuracy: 0.9667 - val_loss: 14.6454 - val_accuracy: 0.9683

===========================] - 13s 7ms/step - loss: 0.0778 - accuracy: 0.9766 - val_loss: 12.6592 - val_accuracy: 0.9728

===========================] - 11s 6ms/step - loss: 0.0586 - accuracy: 0.9816 - val_loss: 13.2272 - val_accuracy: 0.9740

===========================] - 8s 4ms/step - loss: 0.0448 - accuracy: 0.9858 - val_loss: 13.9546 - val_accuracy: 0.9769
```

Figure 2 Evaluation and performance of the model

- **Epoch 1**: Training accuracy: 0.9261 and Validation accuracy: 0.9606
- **Epoch 2**: Training accuracy: 0.9667 and Validation accuracy: 0.9683
- **Epoch 3**: Training accuracy: 0.9766 and Validation accuracy: 0.9728
- **Epoch 4**: Training accuracy: 0.9816 and Validation accuracy: 0.9740

After completing the training process, the model was evaluated on the test dataset to determine its performance on unseen data. The test accuracy achieved by the model was:

3.5. Validation Test Accuracy: 0.9769

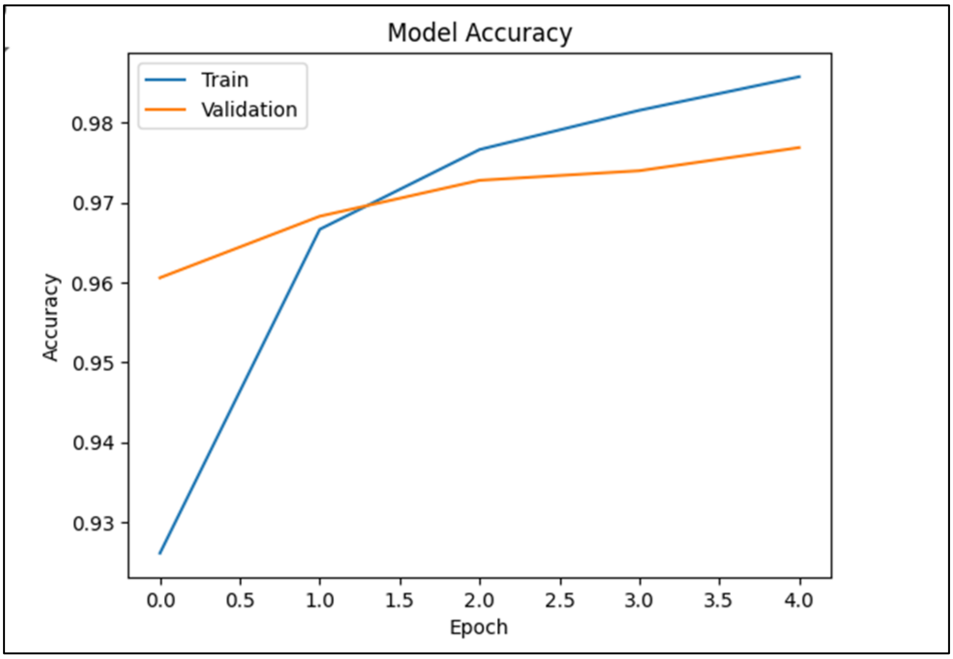

Figure 3 Learning Curve showing the increase in accuracy as the number of epochs increases

In the context of our project, the learning curve serves as a vital tool to visualize and understand the training dynamics of our deep learning model. A learning curve plots the model's performance metrics, such as accuracy (as used in this project), against the number of epochs, providing insights into how well the model is learning from the data over time.

For our project, we have observed that both the training and validation accuracy reach high levels, approximately **98%,** and continue to increase steadily as the number of epochs progresses. This indicates a well-performing model with stable and improving accuracy metrics.

3.6. Key Observations from the Learning Curve

- **High Accuracy:** Both training and validation accuracies are consistently high, reaching around 98%. This suggests that our model effectively learns and generalizes from the data.
- **Continuous Improvement:** The continuous increase in both training and validation accuracies as the number of epochs increases indicates that the model benefits from additional training without signs of overfitting or underfitting. This steady improvement is a positive sign that our preprocessing techniques and model architecture are robust and well-suited to the task.
- **Stability:** The absence of significant fluctuations in the learning curve demonstrates model stability. This stability implies that the model maintains its performance across different epochs, reinforcing the reliability of our approach.

These results indicate that the model was able to learn and generalize well from the training data, achieving a high level of accuracy on both the training and test datasets. This demonstrates the effectiveness of the preprocessing techniques and the data augmentation strategies used in preparing the data, as well as the suitability of the chosen neural network architecture for the task of handwritten digit recognition.

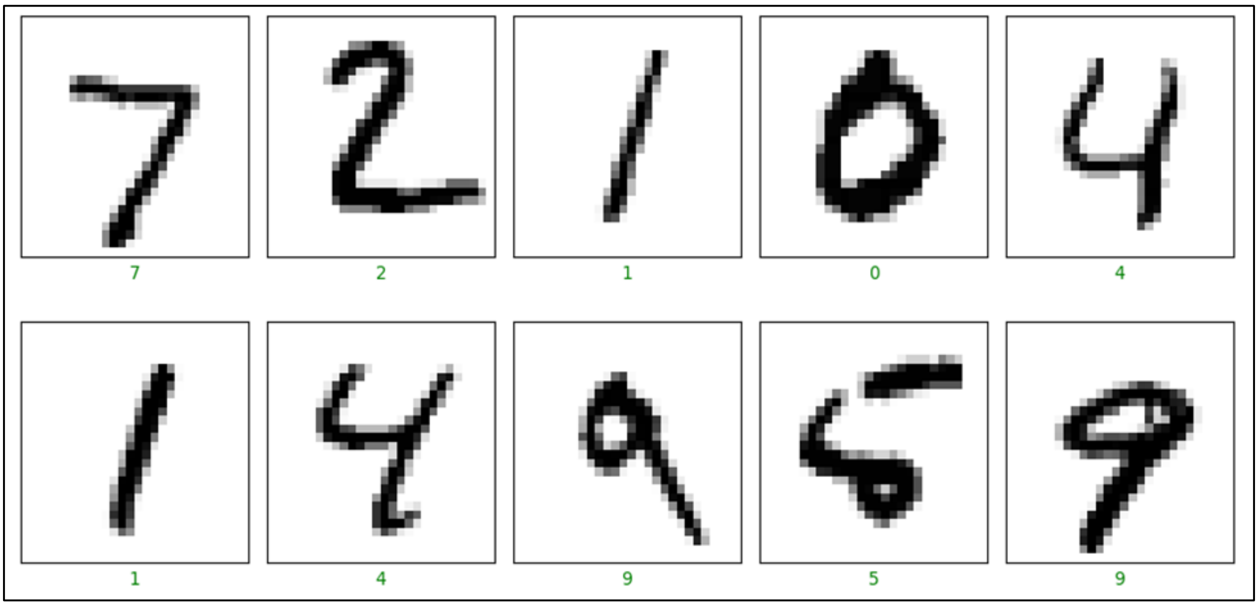

Figure 4 Image showing the accurate prediction of the model sample written numbers

In this methodology, we successfully built and trained a neural network model for handwritten digit recognition using the MNIST dataset. By employing data normalization and augmentation techniques, we enhanced the diversity and quality of the training data, leading to improved model generalization. The model architecture, consisting of a flattening layer, a hidden dense layer with ReLU activation, and an output dense layer with softmax activation, proved effective in capturing the features necessary for accurate digit classification.

After training for three epochs, the model achieved impressive accuracy improvements, culminating in a validation accuracy of 0.9769. This high level of performance underscores the robustness of our preprocessing and augmentation strategies, as well as the efficacy of the neural network design. Overall, this methodology provides a solid foundation for more complex handwritten text recognition tasks and can be further refined to handle more diverse and intricate datasets.

3.7. Applications of Handwritten Text Recognition

Handwritten Text Recognition (HTR) has a wide range of applications across various industries, leveraging the ability to convert handwritten content into machine-readable text. Here are some of the key areas where HTR is making a significant impact:

- **Digital Archiving and Document Management:** Libraries and archives use HTR to digitize historical manuscripts, books, and documents, making them accessible and searchable online. This helps preserve valuable historical records and facilitates research by providing easy access to digitized texts. In corporate environments, HTR aids in automating the management of handwritten documents such as meeting notes, contracts, and forms, improving efficiency and reducing manual data entry errors.
- **Education and E-Learning**: HTR is utilized in educational settings for automated grading systems that evaluate handwritten assignments and exams, providing quicker feedback to students and reducing the workload on educators. Additionally, HTR enables the development of interactive learning applications where students can write answers by hand, and the system can recognize and assess their responses in real time.
- **Healthcare**: In healthcare, HTR is used to digitize handwritten medical records, prescriptions, and doctor's notes, facilitating the transition to electronic health records (EHR). This enhances the accessibility and readability of patient information, improving overall healthcare service efficiency. Medical researchers can also use HTR to analyze large volumes of handwritten clinical notes and case studies, aiding in the extraction of valuable insights and data for research.
- **Finance and Banking**: Banks use HTR to automate the processing of handwritten checks, reducing the time and effort required for manual verification and entry. Financial institutions leverage HTR to digitize various forms and applications, streamlining the onboarding process and improving customer service.
- **Postal Services and Mail Sorting**: Postal services employ HTR to automatically read and sort handwritten addresses on mail and packages, enhancing the speed and accuracy of mail delivery. HTR also helps in the automatic recognition of handwritten tracking numbers and other details, improving the efficiency of logistics and package tracking systems.
- **Legal Industry**: Legal professionals use HTR to digitize and search through handwritten legal documents, contracts, and notes, streamlining the document review process. HTR assists in organizing and managing handwritten case notes and evidence, making it easier to retrieve and analyze information.
- **Customer Service and Feedback**: Companies use HTR to analyze handwritten customer feedback and surveys, gaining valuable insights into customer opinions and preferences. Handwritten support tickets and complaint forms can be digitized and processed automatically, improving response times and service quality.

4. Conclusion

Handwritten Text Recognition (HTR) technology has proven to be a valuable tool across various industries, enhancing the efficiency, accuracy, and accessibility of handwritten information. The advancements in machine learning and data processing have significantly improved HTR performance, making it an essential component in digitization efforts.

Future direction

Looking forward, the future direction of HTR will likely involve further integration with artificial intelligence, enabling even more accurate and context-aware recognition. Advances in deep learning and neural networks, coupled with the availability of larger and more diverse datasets, will continue to drive improvements. Additionally, expanding HTR applications to handle complex handwriting styles and multilingual documents will broaden its utility and impact across more sectors.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] S. Pavithran, linkdin, 16 February 2024. [Online]. Available: https://www.linkedin.com/pulse/unleashing-power-tensorflow-data-analytics-insights-through-s-d7wnc. [Accessed 20 June 2024].

[2] A. Alaei, Automatic Segmentation of Persian Handwritten Text Enabling Accurate Recognition, Manasagangotri, 2012.

[3] Y. Chajri and B. Bouikhalene, Recognition of Handwritten Mathematical Text, International Journal of Future Generation Communication and Networking, vol. 9, no. 8, pp. 307-316, 2016.

[4] T. Rampure, H. Damahe, K. Bharati, D. Raj and N. Naikwade, Handwritten Text Recognition, International Journal of Advanced Research in Science Communication and Technology, vol. 4, no. 4, pp. 774-779, 2024.

[5] Q. Zheng and Y. Zhang, Text Detection and Recognition for X-ray Weld Seam Images, Applied Science, vol. 14, no. 2422, pp. 1-17, 2024.

[6] Y. Wang, Y. Du, Z. Pang, J. Qi and H. Ge, The Text Detection and Recognition Method for Electrical Nameplates Based on Deep Learning, Electronic Engineering and Informatics, pp. 589-598, 2024.

[7] J. Liang and Y. Qi, Application of Artificial Intelligence Technology in Text Recognition and Detection Algorithms, in Frontier Computing on Industrial Applications, Singapore, 2024.