Journal of **N**onlinear **A**nalysis and **O**ptimization
Vol. 16, Issue. 1: 2025
ISSN : **1906-9685**

Live Handwritten Digit and Character Recognition Prediction Using Deep Learning

Dayakar Tammineni 1,N.V.SOUNDARYA[1]

#1Associate Professor, Dept of CSE,PBR VITS,KAVALI

#1PG scholar , Dept of CSE,PBR VITS,KAVALI

Abstract: Using cutting-edge machine learning frameworks, the Handwritten Recognition system created for this research provides a high-accuracy solution for recognising both letters and numbers. Utilising the MNIST and EMNIST datasets, the system primarily uses Convolutional Neural Networks (CNNs) for digit identification and Residual CNNs for character recognition. Batch normalisation, dropout, and advanced optimisation techniques are used to fine-tune these models in order to guarantee their resilience and flexibility under a variety of input situations. Dynamic learning is made possible by the system's integration of custom dataset management, which enables user-corrected data to gradually improve and customise forecasts. By expanding on user contributions, this adaptive method tackles one of the most prevalent problems in handwriting recognition: managing distinctive and non-standard handwriting styles.

The project uses a web-based interface with an interactive canvas for real-time input to make it accessible and easy to use. Users may immediately draw characters or numbers, get predictions right away, and make changes as needed. Preprocessing pipelines, which standardise and normalise inputs, further improve the predictions and guarantee model compatibility even in difficult situations like handwriting that is noisy or fragmentary. Additionally, the deployment infrastructure allows for seamless use across several platforms by combining Pyngrok for remote accessible and HTTP Server for localised hosting. This project's emphasis on performance and usability not only results in a potent handwriting recognition solution, but it also establishes a scalable framework for future recognition of complicated inputs, such entire words or phrases. This system's scalability, interactivity, and efficiency put it in a good position to contribute significantly to areas including accessibility technology, document digitisation, and education.

Index terms - *Handwritten recognition; deep learning; CNN; residual networks; MNIST; EMNIST; real-time prediction; adaptive learning; character recognition; digit recognition; preprocessing; web*

interface; user interaction; document digitization; accessibility.

1. INTRODUCTION

By bridging the gap between the ordered accuracy of digital systems and the organic, freeform character of human writing, handwritten recognition technology is a game-changer that allows machines to comprehend and handle one of the earliest forms of human communication. This research uses deep learning to address a variety of real-world problems by introducing a sophisticated, dual-purpose handwriting recognition system that is painstakingly developed to recognise both alphabetic characters (A-Z, a-z) and numerical digits (0-9). The system's two primary neural network designs are Residual Convolutional Neural Networks (ResNets) for character recognition and Convolutional Neural Networks (CNNs) for digit recognition. The CNN model, specifically designed for digits, consists of five convolutional layers with filter sizes increasing from 32 to 128; these layers are separated by max-pooling operations and end with dense layers, which are optimised to efficiently categorise the 10 different digit classes.

With seven convolutional layers with filter sizes up to 512 and residual connections that help it overcome the vanishing gradient issue, the ResNet model, on the other hand, is made for characters and can distinguish the nuances of 37 alphabetic classes. The benchmark datasets used to train these models are well-known in the machine learning community: MNIST, which offers 10,000 test and 60,000 training images of 28x28 greyscale digits, and EMNIST ByMerge, which offers a varied sample of handwritten letters by filtering out digits and concentrating on 37 alphabetic characters. From the organised simplicity of numerical strokes to the complex flourishes of alphabetic letters, our dual-dataset technique guarantees that the system catches the whole range of handwriting variations.

In the fields of education, finance, and healthcare, it can be used to digitise handwritten notes or help grade assignments; in human-computer interaction, it can improve accessibility for touch-based devices like tablets or smartboards; and in healthcare, it can be used to transcribe doctor's prescriptions. This project integrates both capabilities, covering a wider range of use cases without compromising performance, in contrast to many current systems that just concentrate on letters or numbers. The high accuracy benchmark of the MNIST dataset—which frequently surpasses 99% with CNNs—is used as a target for digits, whilst the intricacy of EMNIST forces the system to attain similar precision for characters, which is verified using pre-trained weights with retraining options if necessary. This system redefines the practical utility of handwritten recognition by combining state-of-the-art deep learning, an interactive web interface, and remote deployment. It provides a flexible and scalable solution for individuals, businesses, educators, and others.

2. LITERATURE SURVEY

a) Handwritten Digit Recognition using Machine and Deep Learning Algorithms:

https://www.researchgate.net/publication/343054680_Handwritten_Digit_Recognition_using_Machine_and_Deep_Learning_Algorithms

Humans have never been more dependent on technology thanks to the development of deep learning and machine learning algorithms, which can now recognise objects in photos and add sound to silent films. Handwritten text recognition research and development is another important and exciting field. Handwriting recognition (HWR), sometimes known as handwritten text recognition (HTR), is one method by which computers can decipher handwritten data [1]. I believe this study uses CNN, MLP, and SVM models to recognise handwritten numbers in MNIST datasets. This study assesses the accuracy and execution time of the aforementioned models to identify the optimal digit recognition model.

b) Automatic prediction of age, gender, and nationality in offline handwriting:

https://www.researchgate.net/publication/265592404_Automatic_prediction_of_age_gender_and_nationality_in_offline_handwriting

There are several reasons why handwriting is categorised according to gender, age, and nation of origin. Forensic investigators can identify certain authors with the use of handwriting categorisation. Few studies have been conducted on this topic. Feature extraction and classification are steps in the handwriting demographic categorisation process. Characterising features makes it possible to identify authors since feature extraction affects system performance. In order to characterise and classify handwritings by age, gender, and nation, we give a wide range of geometric criteria in this work. Both random forests and kernel discriminant analysis make advantage of feature fusion. In the QUWI dataset, the classification rates for gender, age range, and nationality are 74.05%, 55.76%, and 53.66%, respectively, when all writers utilise the same handwritten text. The percentages, however, decrease to 73.59%, 60.62%, and 47-98%, respectively, when authors employ different styles.

c) Handwritten Recognition Using SVM, KNN and Neural Network:

https://www.researchgate.net/publication/313247443_Handwritten_Recognition_Using_SVM_KNN_and_Neural_Network

Computers can interpret handwritten text from a range of media, such as paper, photos, and touch screens, by using handwriting recognition (HWR). This article will teach us how to use SVM, KNN, and a neural network to recognise handwriting.

d) The Recognition of Handwritten Digits Based on BP Neural Network and the Implementation on Android:

https://ieeexplore.ieee.org/document/6455316

Offline handwriting recognition is a popular topic in pattern recognition and image processing, with a wide range of applications such as mail sorting, cheque recognition, assistive reading devices for the blind, and more. This study uses Back Propagation (BP) neural networks to identify handwritten numbers using feature extraction; the MNIST handwritten digit database is used for training and testing the neural network; we also propose Principal Component Analysis (PCA) for feature extraction, w

e) Backpropagation Applied to Handwritten Zip Code Recognition:

https://ieeexplore.ieee.org/document/6795724

This research shows how to incorporate such constraints into a backpropagation network, which has been used by the USPS to recognise handwritten zip code digits. In this approach, a single network learns the entire recognition process from character normalisation to classification. Learning networks are better able to generalise when given specific tasks to complete.

3. METHODOLOGY

A. Proposed Work:

The suggested system is a strong and clever framework that combines deep learning models with a self-learning process through customised data handling for real-time handwritten character and digit detection. It addresses a broad range of use cases, including assistive technology, digital forms, and instructional applications, when handwriting inputs differ significantly amongst users.

The system ensures full alphanumeric recognition capabilities by integrating the MNIST and EMNIST ByMerge datasets, which enable comprehensive recognition of letters (A–Z, a–z) and numbers (0–9). The Convolutional Neural Network (CNN) architecture at the heart of the system is strengthened by residual connections, which enables it to efficiently learn intricate patterns from the input pictures while avoiding problems like disappearing gradients. To capture different writing styles and character distortions, this deep network is trained on extensive datasets. The residual structure improves identification accuracy by stabilising the training process and deepening learning. With an accuracy of 99.64%, the digit model trained on the MNIST dataset demonstrated its efficacy. Users may rapidly amend misclassified samples thanks to the system's real-time correction function. These updated inputs are saved in a unique dataset and then utilised again to improve feature-based similarity predictions in the future. The system is adaptable, scalable, and prepared for incorporation into practical applications thanks to the mix of deep learning, user input, and modular architecture.

B. System Architecture:

The architecture of the proposed system is built around a deep learning backbone, primarily leveraging Convolutional Neural Networks (CNNs) for digit recognition and Residual CNNs for character recognition. The input layer accepts raw images drawn on an interactive canvas, which are then passed through a preprocessing pipeline that normalises and resizes them to match the expected input format of the models. The CNN-based model processes these images, extracting essential features through multiple convolutional and pooling layers. Batch normalization and dropout techniques are applied to improve training stability and avoid overfitting. For character recognition, residual blocks allow deeper learning by maintaining gradient flow, enabling the system to handle more complex patterns and varied handwriting styles.

The system is designed with modularity and interactivity at its core. A web-based interface allows users to draw characters or digits, which are instantly processed and predicted. If a prediction is incorrect, users can correct the output, and the system stores this corrected data in a custom dataset for future training cycles. This enables continuous improvement via adaptive learning. The application is hosted using

HTTP Server for local access and Pyngrok for global remote access, ensuring flexibility in deployment. This real-time, feedback-driven architecture ensures not only accurate and efficient recognition but also supports scalability for future expansions, such as recognizing full words or cursive handwriting.

Fig 1Proposed Architecture

C. Modules:

Module Name	Module Description
Canvas Module	Captures user-drawn input on a 280x280 canvas and sends the image data to the backend for processing.
Preprocessing Module	Applies image resizing, normalization, erosion, and thresholding to convert input into a model-ready format.
Digit Model Module	Uses a CNN model trained on the MNIST dataset to predict digits (0–9) with high accuracy (up to 99.64%).
Character Model Module	Uses a ResNet-based model trained on the EMNIST ByMerge dataset to predict characters (A–Z, a–z).
Prediction Handler	Handles prediction requests, processes model outputs, and matches with custom dataset using feature similarity.
Correction Module	Accepts user-corrected inputs and saves them into a custom HDF5 dataset for improved future predictions.
Custom Dataset Module	Manages HDF5 files (custom_digits.h5, custom_characters.h5) and extracts features for similarity matching.

Web Interface Module	Provides interactive buttons, canvas drawing, and result display via HTML, CSS, and JavaScript.
Server Module	Python-based backend using HTTP Server, handles /predict and /correct endpoints.
Ngrok Deployment Module	Exposes the local server publicly via Ngrok tunnel to allow real-time access across networks.

D. Algorithms

a) **Convolutional neural networks (CNNs):** A CNN defeated a human in an object recognition test for the first time in 2015. Convolutional neural networks (CNNs), which are mostly employed in computer vision and image classification applications, are able to identify characteristics and patterns in an image, allowing tasks like object identification or recognition.

b) **Residual CNN Architecture (ResNet) :** Character recognition using the Residual CNN Architecture (ResNet), trained on EMNIST ByMerge, is intended to manage deeper layers and reduce disappearing gradients.

4. EXPERIMENTAL RESULTS

Accuracy: The ability of a test to differentiate between healthy and sick instances is a measure of its accuracy. Find the proportion of analysed cases with true positives and true negatives to get a sense of the test's accuracy. Based on the calculations:

Accuracy = TP + TN /(TP + TN + FP + FN)

$$Accuracy = \frac{(TN + TP)}{T}$$

Precision: The accuracy rate of a classification or number of positive cases is known as precision. Accuracy is determined by applying the following formula:

Precision = True positives/ (True positives + False positives) = TP/(TP + FP)

$$\mathrm{Pr}\, e\, cision = \frac{TP}{(TP + FP)}$$

Recall: The recall of a model is a measure of its capacity to identify all occurrences of a relevant machine learning class. A model's ability to detect class instances is shown by the ratio of correctly predicted positive observations to the total number of positives.

$$Recall = \frac{TP}{(FN + TP)}$$

mAP: One ranking quality statistic is Mean Average Precision (MAP). It takes into account the quantity of pertinent suggestions and where they are on the list. The arithmetic mean of the Average Precision (AP) at K for each user or query is used to compute MAP at K.

$$mAP = \frac{1}{n}\sum_{k=1}^{k=n} AP_k$$

$$AP_k = \text{ the AP of class } k$$

$$n = \text{ the number of classes}$$

F1-Score: A high F1 score indicates that a machine learning model is accurate. Improving model accuracy by integrating recall and precision. How often a model gets a dataset prediction right is measured by the accuracy statistic..

$$F1 = 2 \cdot \frac{(Recall \cdot \Pr e\, cision)}{(Recall + \Pr e\, cision)}$$

Fig 2. Home Page

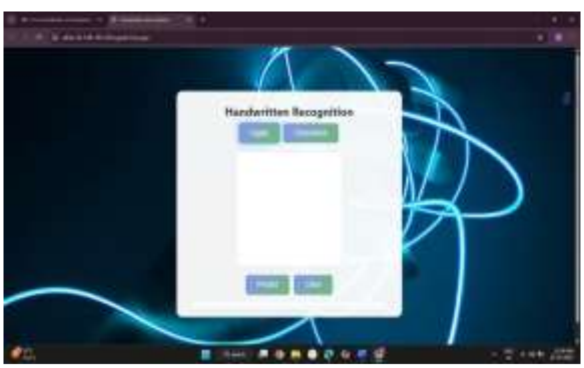

Fig 2. Prediction Page

Fig 2. Predicted results

5. CONCLUSION

By combining two deep learning models—one for digits using the MNIST dataset and another for characters using the EMNIST ByMerge dataset—the developed handwritten digit and character recognition system provides an accurate and adaptable solution that can be used for real-world applications like automated form reading, exam processing, and educational tools. The system's backend is built to handle real-time prediction through a centralised server, and its modular structure ensures maintainability and scalability. One of the system's main strengths is its capacity to learn from misclassifications; it has a correction mechanism that enables users to provide the correct label when a prediction is incorrect, and stores these corrected samples in a custom dataset that the system uses for feature-based matching to improve future predictions.

6. FUTURE SCOPE

By using scripts such as Devanagari, Tamil, Arabic, or Chinese to train models, the system can be expanded to support multilingual handwriting recognition. Additionally, it can offer real-time input suggestions, assisting users in correcting unclear drawing strokes. It is possible to create an analytics backend dashboard that displays model performance, adjustments, and forecast trends. For offline use, the system could be made into a mobile application with ONNX or TensorFlow Lite. Few-shot learning might be used to personalise handwriting, allowing the system to adjust to each user's preferences. For more precise predictions, advanced natural language processing (NLP) can add linguistic context. Multimodal input may be possible through voice-to-text integration. Handwritten forms can be digitised with the use of integration with document scanners. Cloud storage might be utilised for synchronising user corrections across devices. Finally, security measures like user authentication may be incorporated to secure bespoke datasets and tailored forecasts.

REFERENCES

[1] Mayank Jetal, "Handwritten Digit Recognition Using CNN,"doi: 10.1109/ICIPTM52218.2021.9388351, IEEE Access, 2021.

[2] Saqib Aetal, "A robust CNN model for handwritten digits recognition and classification," doi: 10.1109/AEECA49918.2020.9213530, IEEE Access, 2021.

[3] Simon Cetal, "Integrating Writing Dynamics in CNN for Online Children Handwriting Recognition," doi: 10.1109/ICFHR2020.2020.00057, IEEE Access, 2020.

[4] Chao Zetal, "Handwritten Digit Recognition Based on Convolutional Neural Network," doi: 10.1109/CAC51589.2020.9326781, IEEE Access, 2021.

[5] Durjoy S Metal, "CNN based common approach to handwritten character recognition of multiple scripts," doi: 10.1109/ICDAR.2015.7333916, IEEE Access, 2015.

[6] Zhiqi Y; Kai F, "Design and implementation of handwritten digit recognition system based on template method," doi: 10.1109/IAEAC.2018.8577760, IEEE Access, 2018.

[7] En M Cetal, "Handwritten Character and Digit Recognition with Deep Convolutional Neural Networks: A Comparative Study," doi: 10.1109/ICoICT58202.2023.10262721, IEEE Access, 2023.

[8] Ayush K Aetal, "A Robust Model for Handwritten Digit Recognition using Machine and Deep Learning Technique," doi: 10.1109/INCET51464.2021.9456118, IEEE Access, 2021.

[9] Caiyun M; Hong Z, "Effective handwritten digit recognition based on multi- feature extraction and deep analysis," doi: 10.1109/FSKD.2015.7381957, IEEE Access, 2016.

[10] Jie M et al, "Handwritten digits recognition based on improved label propagation algorithm," doi: 10.1109/ICCOINS.2016.7783239, IEEE Access, 2016.

[11] Eshika J; Amanveer S, "Efficient Handwritten Digit Classification using Convolutional Neural Networks: A Robust Approach with Data Augmentation," doi: 10.1109/ICACRS62842.2024.10841545, IEEE Access, 2025.

[12] Subharathna N et al, “Decoding Handwritten Characters using Convolutional Neural Networks (CNNs),” doi: 10.1109/ICSCSS60660.2024.10625226, IEEE

Access, 2024.

[13] Chandradeep B et al, “Deep Learning for Handwritten Character Recognition,” doi: 10.1109/ICSEIET58677.2023.10303610, IEEE Access, 2023.

[14] Yawei H; Huailin Z, “Handwritten digit recognition based on depth neural network,” doi: 10.1109/ICIIBMS.2017.8279710, IEEE Access, 2018.

[15] Mahmoud M A G; Ashraf A Y M, “A Comparative Study on Handwriting Digit Recognition Using Neural Networks,” doi: 10.1109/ICPET.2017.20, IEEE Access, 2017.

[16] Gregory Cetal, “EMNIST: Extending MNIST to handwritten letters,” doi: 10.1109/IJCNN.2017.7966217, IEEE Access, 2017.

[17] Peiyu M, “Recognition of Handwritten Digit Using Convolutional Neural Network,” doi: 10.1109/CDS49703.2020.00044, IEEE Access, 2020.

[18] Caihua L et al, “Handwritten character recognition with sequential convolutional neural network,” doi: 10.1109/ICMLC.2013.6890483, IEEE Access, 2013.

[19] Sen F, “Handwritten Digital Detection Based on Tensorflow Building SSD Model,” doi: 10.1109/ICIASE45644.2019.9074012, IEEE Access, 2020.