iJRASET
International Journal For Research in Applied Science and Engineering Technology



# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

**Volume:** 14 **Issue:** III **Month of publication:** March 2026

**DOI:** https://doi.org/10.22214/ijraset.2026.78507

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
*ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538*
*Volume 14 Issue III Mar 2026- Available at www.ijraset.com*

# Real-Time Handwritten Digit, Character and Mathematical Symbol Recognition Using Deep Learning

Dr. P.V.N. Rajeswari[1], N. Anusha[2]
[1]*Associate professor,* [2]*PG Scholar, Department of CSE, PBR Visvodaya Institute of Technology and science, kavali, A.P, India.*

***Abstract:*** ***Handwritten recognition of mathematical content is a challenging task due to diverse writing styles and the coexistence of digits, alphabets, and mathematical symbols. This project presents a unified deep learning–based system for handwritten digit, character, and mathematical symbol recognition using a multi-model architecture. The proposed framework employs Convolutional Neural Networks (CNNs) trained on standard benchmark datasets, including MNIST for digit recognition, EMNIST for handwritten character recognition, and the CROHME handwritten mathematical symbol dataset for mathematical symbol classification. The architecture combines TensorFlow/Keras-based models for digit and character recognition with a PyTorch-based optimized CNN for mathematical symbol recognition.***
***A robust image preprocessing pipeline is designed to normalize handwritten inputs, enhance contrast, and preserve structural features critical for accurate recognition. To improve adaptability and robustness, the system incorporates a user FEEDBACK–driven correction mechanism, where misclassified samples are stored in custom datasets and matched using deep feature similarity measures during inference. This feedback-driven approach allows the system to improve recognition accuracy over time without full retraining. The system further incorporates ensemble prediction strategies and cosine similarity–based feature matching to enhance robustness.***
***Keywords: Convolutional Neural Networks (CNNs)****, Deep learning,* ***EMNIST, MNIST, CROHME***

## I. INTRODUCTION

Handwritten recognition represents a transformative technology that bridges the gap between the organic, freeform nature of human writing and the structured precision of digital systems, enabling machines to interpret and process one of humanity's oldest forms of communication. This project introduces an advanced handwritten recognition system meticulously designed to identify numerical digits (0– 9), alphabetic characters (A–Z, a–z), and individual mathematical symbols, harnessing the power of deep learning to tackle a wide array of real-world challenges. At its core, the system employs multiple specialized neural network architectures: Convolutional Neural Networks (CNNs) for digit recognition, Residual Convolutional Neural Networks (ResNets) for character recognition, and a dedicated convolutional model for mathematical symbol classification.

### A. *Methodology*

The methodology of this project adopts a modular and systematic approach to develop a unified handwritten recognition system capable of identifying handwritten digits, alphabetic characters, and individual mathematical symbols. Separate deep learning models are designed and trained for each recognition category to ensure high accuracy and robustness.

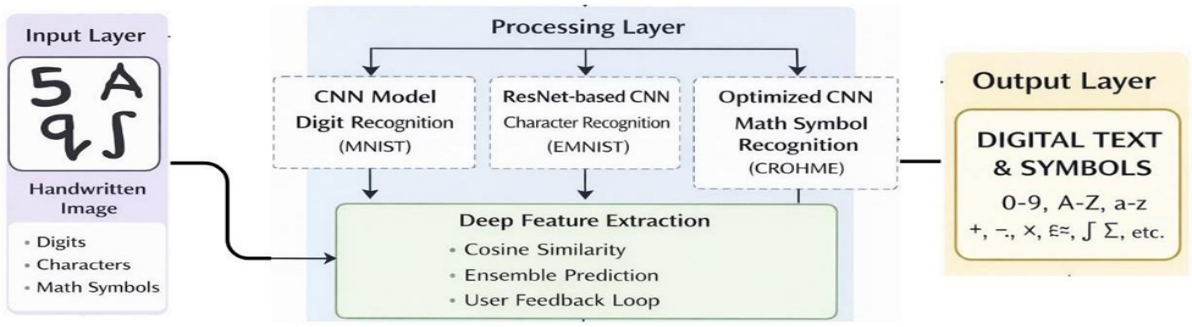



Figure. 1. Handwritten Recognition System Workflow

©IJRASET: All Rights are Reserved 

## II. EXISTING SYSTEM

The existing system focuses on recognizing handwritten digits (0–9) using Artificial Neural Networks (ANNs) trained with a back-propagation algorithm. The input images are preprocessed through binarization, resizing to 16×16 pixels, and thinning to obtain skeletonized forms. Feature extraction is performed using horizontal, vertical, and diagonal histograms, which are concatenated to form a feature vector. This vector is provided as input to a three-layer neural network consisting of 94 input units, 15 hidden units, and 10 output units. Experimental results show that the system can achieve high accuracy, reaching up to 99% under controlled conditions.

## III. PROPOSED SYSTEM

The proposed system is a robust and intelligent framework designed for real-time handwritten recognition of digits, alphabetic characters, and individual mathematical symbols using deep learning techniques. The system employs multiple convolutional neural network–based models trained on standard datasets. A CNN trained on the MNIST dataset is used for handwritten digit recognition and achieves an accuracy of 99.64%, while a deeper CNN with residual connections trained on the EMNIST ByMerge dataset enables accurate recognition of alphabetic characters (A–Z, a–z). In addition, a dedicated CNN trained on samples derived from the CROHME dataset is used for handwritten mathematical symbol recognition.

## IV. LITERATURE REVIEW

The experimental results confirmed the superior performance of the CNN-based model, achieving an accuracy rate exceeding 98.5% on the test dataset. Compared to traditional machine learning models like Decision Trees, Naive Bayes, and k-Nearest Neighbors (k-NN), the CNN outperformed in terms of both accuracy and computational efficiency. The authors noted that the deep hierarchical structure of CNNs enabled the extraction of abstract and invariant features from raw pixel data, which classical models failed to generalize effectively. Furthermore, the study suggested that integrating advanced data augmentation techniques such as rotation, scaling, and shifting could enhance the model's robustness against diverse handwriting patterns. The paper concludes by highlighting the relevance of CNNs in real-world applications including cheque processing, automated banking, digital form recognition, and postal address digitization. Future work includes expanding the system to multilingual digit datasets and hybridizing CNNs with Recurrent Neural Networks (RNNs) to handle continuous handwriting streams, thereby broadening its applicability to cursive script recognition.

## V. SYSTEM DESIGN

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

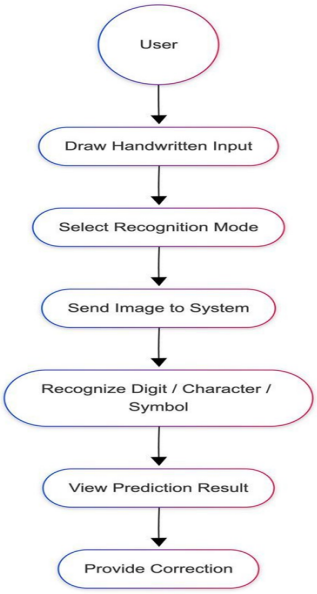



Fig. 2. Use case diagram

A collaboration diagram groups together the interactions between different objects. The interactions are listed as numbered interactions that help to trace the sequence of the interactions. The collaboration diagram helps to identify all the possible interactions that each object has with other objects

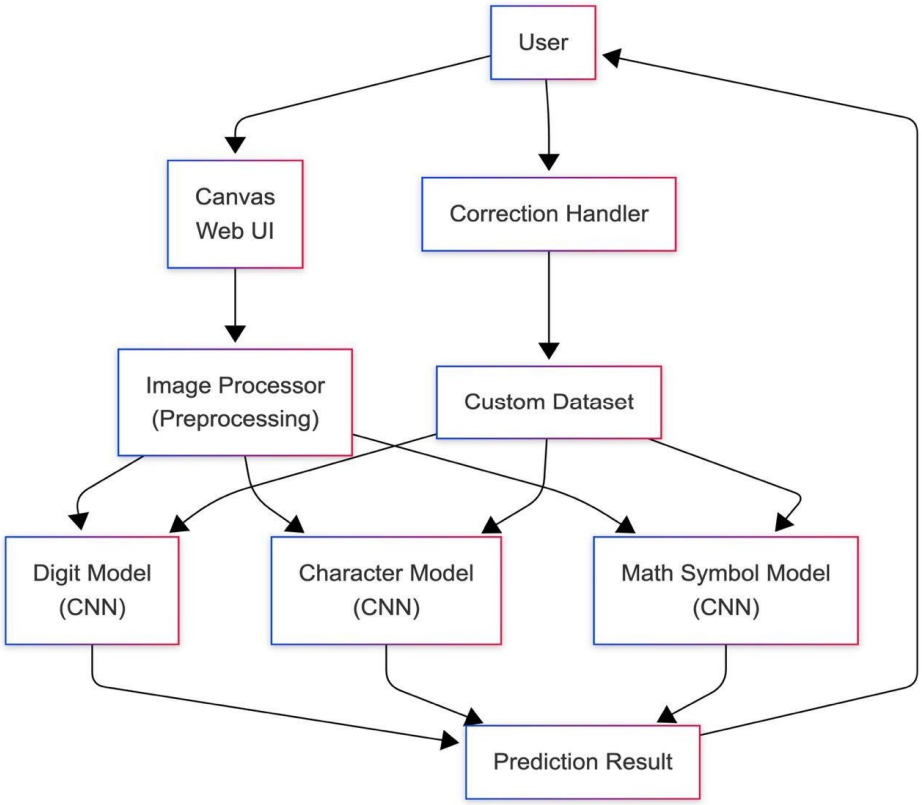



Figure. 3. Collaboration diagram

A sequence diagram represents the interaction between different objects in the system. The important aspect of a sequence diagram is that it is time-ordered

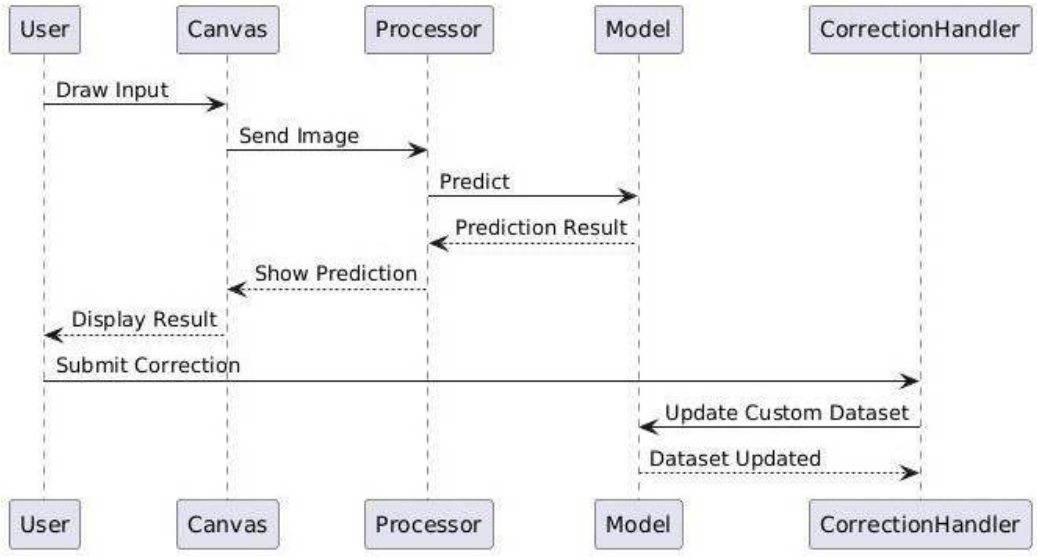



Fig. 4. Sequence diagram

## VI. TESTING

### A. *Testing Strategy*

Testing plays a crucial role in the success of any intelligent recognition system, particularly when it involves user input and real-time prediction. In this handwritten digit and character recognition system, various layers of testing were conducted to ensure the system performs reliably, efficiently, and accurately under different conditions. The system's predictive core, data handling pipeline, correction logic, and user interface were all rigorously validated through structured testing strategies. Both component-level and system-level testing were performed to ensure smooth integration of handwritten digit, character, and mathematical symbol recognition.

*1)* Unit testing
*2)* Integration testing
*3)* Performance testing
*4)* Robustness testing
*5)* Stress testing
*6)* Functional testing
*7)* Error handling testing

### B. *Test Cases*

| Test Case ID | Test Case Description | Input | Expected Output | Status |
|---|---|---|---|---|
| TC_01 | Load digit model successfully | Load improved1.weights.h5 | Model loads successfully | Pass |
| TC_02 | Load character model successfully | Load checkpoint. Weights .h5 | Character model initialized correctly | Pass |
| TC_03 | Load math model successfully | optimized. pth | Math model loads without error | Pass |
| TC_04 | Base64 image preprocessing | se64 PNG string | Normalized grayscale image | Pass |
| TC_05 | Predict digit from user sketch | Drawn digit '5' | Prediction = 5 with confidence > 90% | Pass |
| TC_06 | Predict character from user sketch | Drawn character 'B' | Prediction = B with high confidence | Pass |
| TC_07 | edict math symbol | Drawn '+' | Prediction = '+' | Pass |
| TC_08 | Save corrected digit to custom dataset | Label = 8, image = drawn 8 | Custom entry saved to custom_digits.h5 | Pass |
| TC_09 | Save corrected character to custom dataset | Label = 'G', image = drawn G | Custom entry saved to custom_characters.h5 | Pass |
| TC_10 | Predict from custom dataset | Image previously saved | Custom label returned if similarity is high | Pass |
| TC_11 | Handle invalid base64 input | Invalid string | Error message returned, no crash | Pass |
| TC_12 | Handle empty input image | Empty canvas | Error: "Image processing failed" | Pass |
| TC_13 | isplay prediction instantly on UI | Canvas sketch | Result and confidence shown within 0.5 seconds | Pass |
| TC_14 | witch modes (digit/character) on frontend | User clicks mode toggle | Model switch confirmed, mode updates on UI | Pass |

| | | | | |
|---|---|---|---|---|
| TC_15 | Exceed custom dataset size (stress test) | 200+ saved images | System handles with no major lag | Pass |
| TC_16 | Correct misclassified digit with retraining logic | Image = 3, model says 8, label = 3 | Prediction improves to 3 after correction | Pass |
| TC_17 | Feature similarity threshold evaluation | Similar custom entry exists | Prediction from custom if similarity > model | Pass |

Table. 1. Test Cases

## VII. EXPERIMENTAL RESULTS

### A. *Screenshots of execution*

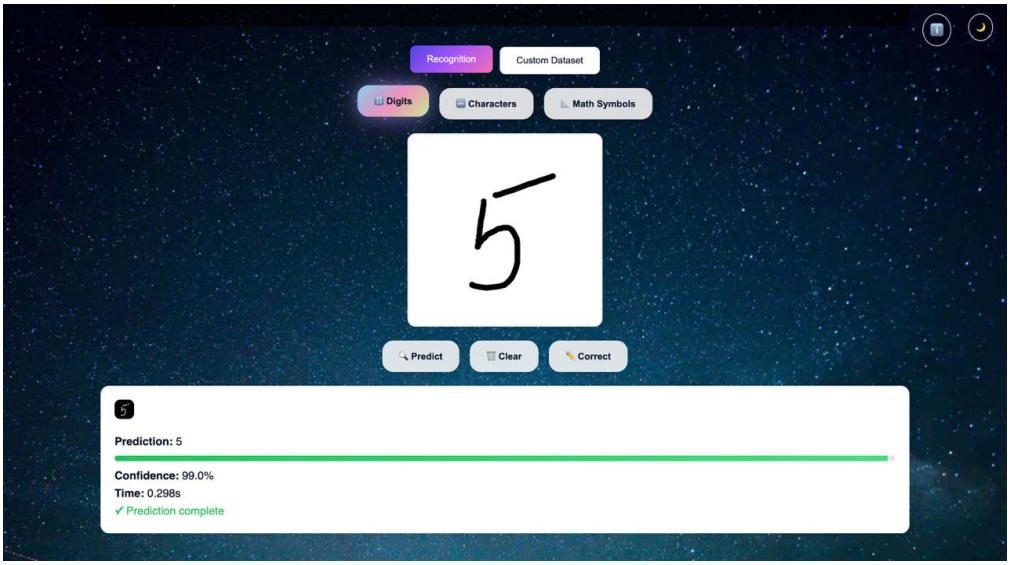



Fig. 5. Incomplete digit

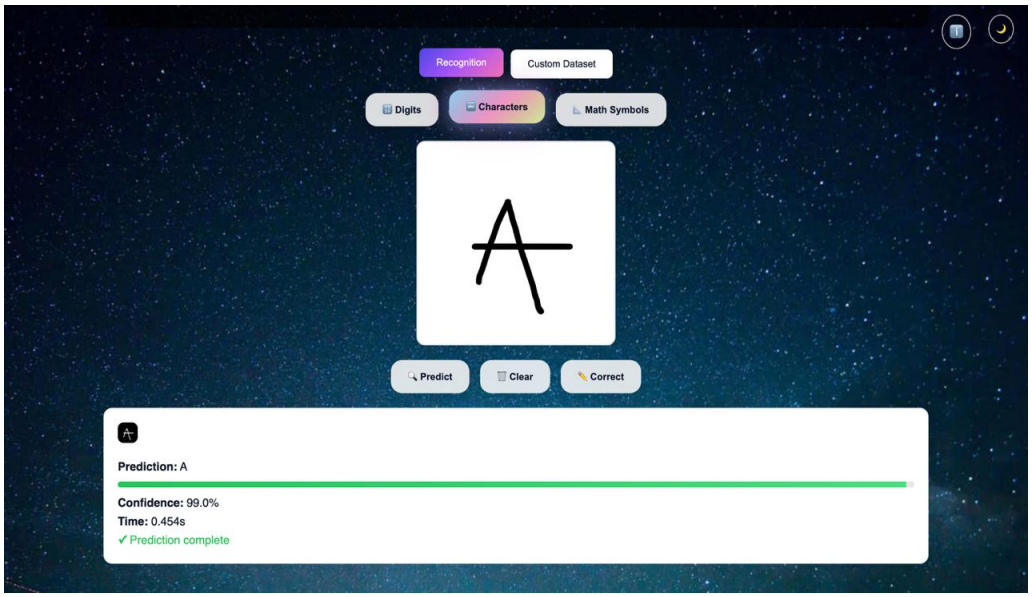



Fig. 6. Character with Uppercase

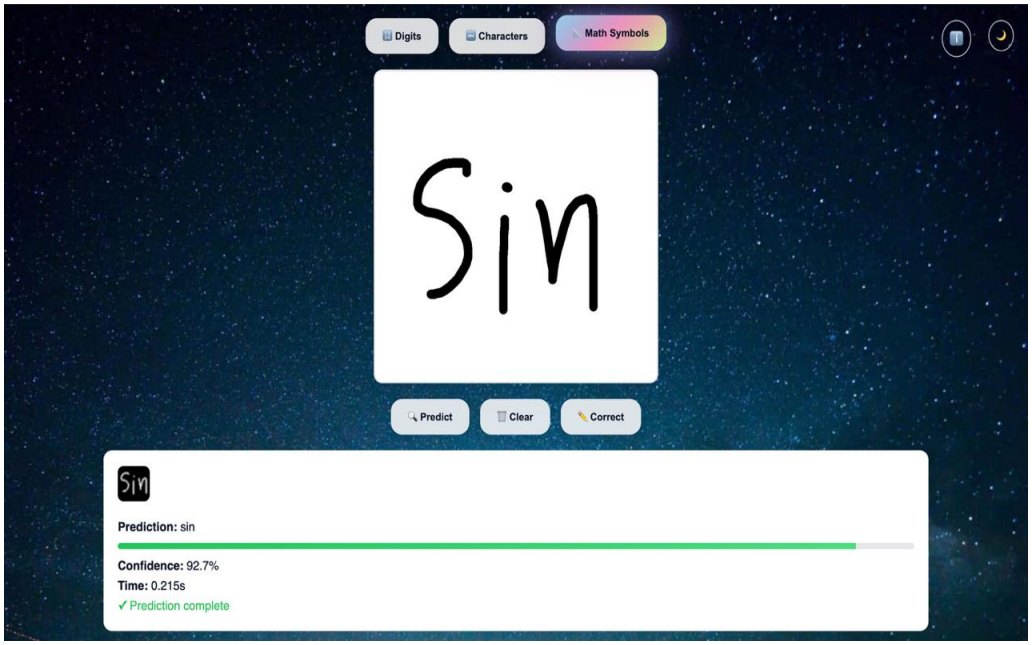



Fig. 7. Mathematical function

## VIII. CONCLUSION AND FUTURE SCOPE

### A. *Conclusion*

The handwritten recognition system successfully delivers an accurate, efficient, and adaptive solution for recognizing handwritten digits, alphabetic characters, and mathematical symbols using deep learning techniques. By integrating a Convolutional Neural Network (CNN) trained on the MNIST dataset for digit recognition, a Residual CNN (ResNet) trained on the EMNIST ByMerge dataset for character recognition, and a dedicated CNN trained on the CROHME dataset for mathematical symbol recognition, the system effectively handles diverse handwriting styles. The real-time web-based interface enables users to draw inputs and receive instant predictions with confidence scores, while the modular backend architecture ensures scalability and maintainability.

### B. *Future scope*

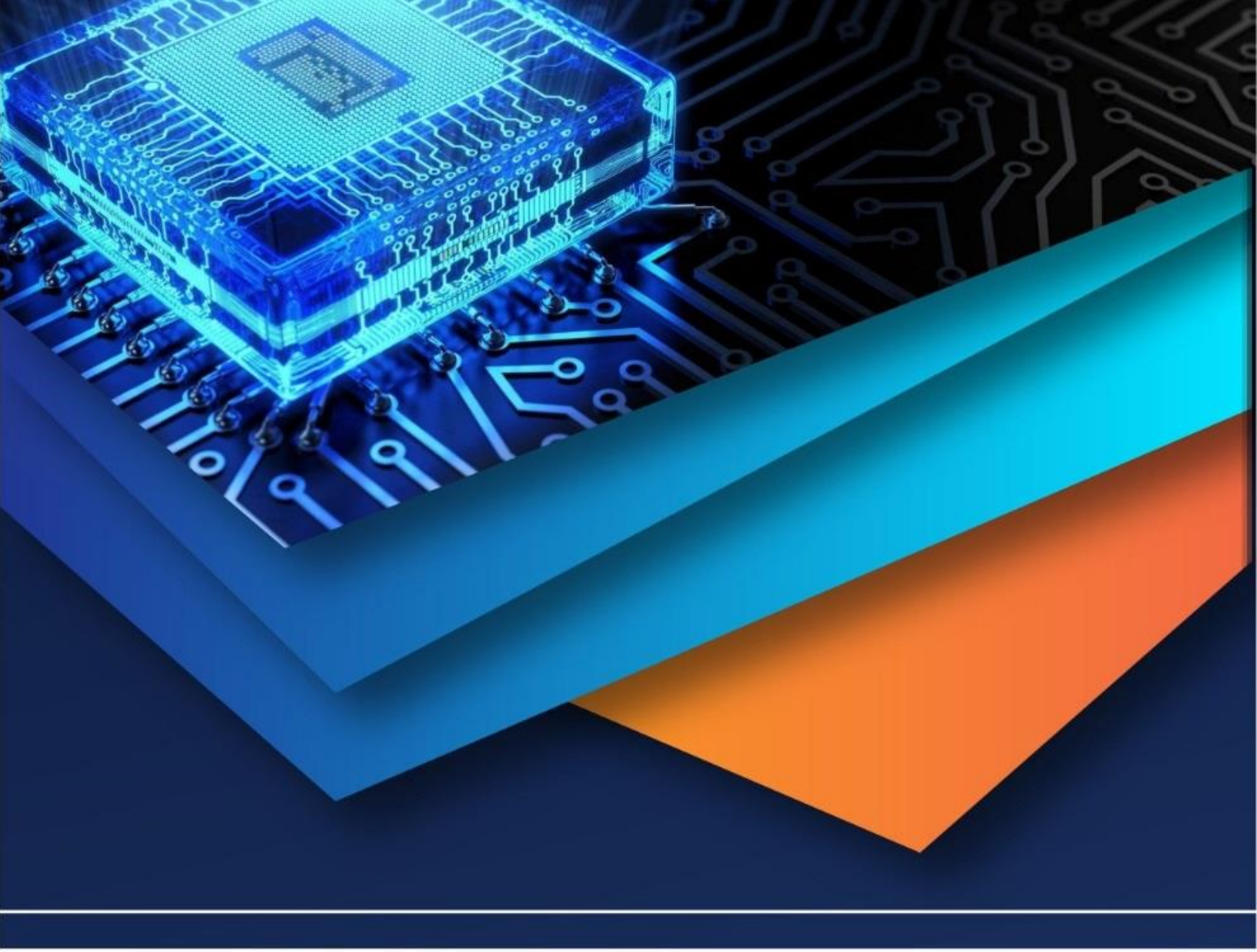
The proposed system can be further enhanced by extending symbol-level recognition to full handwritten mathematical expression recognition, incorporating spatial relationship analysis, structural parsing, and expression evaluation. Future work may also include multilingual handwriting recognition by training models on additional scripts such as Devanagari or Chinese. The system can be deployed as a mobile or cloud-based application to improve accessibility and scalability, while advanced deep learning architectures such as attention mechanisms and transformer-based vision models can be explored to improve accuracy for complex handwriting patterns. Personalized handwriting adaptation using incremental or few-shot learning can enable the system to better adapt to individual users.

## 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.
[20] Chung-Hsing C et al, "Recognition of Handwritten English and Digits Using Stroke Features and MLP," doi: 10.1109/SCISISIS55246.2022.10002071, IEEE Access, 2023.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

# INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)