Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

An Optimized Ensemble Framework for Handwritten Digit Recognition Using CNN and Gradient Boosting Models

[1]Prashant Kumar, [2]Dr. Ragini Shukla
[1]Research Scholar, Department of IT & CS, Dr. C. V. Raman University, Bilaspur, India
[2]Professor, Department of IT & CS, Dr. C. V. Raman University, Bilaspur, India
[1]kool.pk517@gmail.com, [2]raginishukla008@gmail.com

ABSTRACT

Recognizing handwritten digits is a vital part of pattern recognition, playing a key role in applications like digitizing documents, reading license plates, and processing bank checks. In this study, we introduce a new ensemble model that combines Convolutional Neural Networks (CNNs) for extracting image features with advanced boosting classifiers-AdaBoost, XGBoost, and LightGBM-to improve the classification of handwritten English numerals. Using the EMNIST Digits dataset, which includes 70,000 images, our approach begins with preprocessing steps such as normalization and data augmentation. It then uses CNNs to extract deep features from the images, followed by a voting system that blends the outputs of the classifiers for more accurate predictions. We evaluated the model using 10-fold cross-validation and achieved an impressive accuracy of 98.45%-a 1.6% improvement over standalone CNNs. additionally; the system reached a precision, recall, and F1-score of 0.984. This research offers a robust and scalable solution for digit recognition, with promising possibilities for real-time use and expansion into other languages and scripts.

Keywords: Handwritten Digit Recognition, Convolutional Neural Networks, Boosting Classifiers, Ensemble Learning, EMNIST Dataset, Deep Learning

1.INTRODUCTION

Pattern recognition, a fundamental discipline within machine learning, focuses on identifying and classifying patterns in data through feature extraction and classification techniques (Pujari, 2012). Handwritten digit recognition, a key subfield, involves classifying digits (0–9) from diverse sources such as scanned documents, images, or touch screens. This task is inherently challenging due to the significant variability in handwriting styles across individuals, influenced by factors such as writing speed, pen pressure, and cultural differences (Dixit, 2020).

Applications of handwritten digit recognition are vast, including automated mail sorting, license plate recognition, bank check processing, and digitization of historical records, all of which demand high accuracy and robustness. Recent advancements in deep learning, particularly Convolutional Neural Networks (CNNs), have significantly improved recognition performance by extracting complex features from images (Trivedi et al., 2018). However, standalone CNNs often face challenges such as overfitting, sensitivity to noise, and limited generalization on diverse datasets.

Ensemble methods, which combine multiple models to improve performance, offer a promising solution. Boosting classifiers like AdaBoost, XGBoost, and LightGBM have shown efficacy in enhancing classification accuracy by focusing on misclassified samples and optimizing decision boundaries (Boufenar et al., 2018). This paper proposes an optimized ensemble framework that integrates CNN-based feature extraction with gradient boosting classifiers to achieve robust handwritten English numeral recognition using the EMNIST dataset. The framework aims to address the limitations of standalone CNNs by leveraging the strengths of ensemble learning, providing a scalable and accurate solution for real-world applications.

2. RELATED WORK

The field of handwritten digit recognition has seen significant progress with the advent of deep learning and ensemble techniques. Recent studies from 2020–2025 have further advanced the field by introducing novel architectures, optimization strategies, and hybrid models. Table 1 summarizes key studies, highlighting their methodologies, datasets, and achieved accuracies.

Table 1: Summary of Related Work on Handwritten Digit Recognition

Author(s)	Year	Dataset	Methodology	Accuracy
Trivedi et al.	2018	Devanagari Numerals	CNN + Genetic Algorithm	96.54% (200 iterations)
Boufenar et al.	2018	OIHACDB-40, AHCD	CNN with Transfer Learning	100% (OIHACDB-40), 99.98% (AHCD)
Mane et al.	2018	Marathi Numerals	Customized CNN + Softmax	94.93%
Shopon et al.	2017	EMNIST, CMATERDB, ISI	CNN + Blocky Artifact Augmentation	99.56% (EMNIST), 99.83% (CMATERDB), 99.35% (ISI)
Das et al.	2015	Bangla Digits	Two-Pass Soft Computing + SVM	87.26%
Stefano et al.	2013	Multiple Datasets	GA-Based Feature Selection	Improved over baseline
Zhang et al.	2018	ICDAR-2013 Chinese Digits	RNN (LSTM + GRU)	High accuracy (not specified)
Ahlawat & Choudhary	2020	MNIST, EMNIST	CNN with Dropout and Batch Normalization	99.87% (MNIST), 99.53% (EMNIST)
Alom et al.	2021	EMNIST, BanglaLekha	Capsule Networks with Dynamic Routing	99.60% (EMNIST), 98.75% (BanglaLekha)
Gupta & Kumar	2022	Devanagari, EMNIST	Hybrid CNN-RNN with Attention Mechanism	99.65% (Devanagari), 99.58% (EMNIST)
Li et al.	2023	MNIST, Custom Chinese Digits	Vision Transformers with Ensemble Learning	99.92% (MNIST), 99.45% (Chinese Digits)
Sharma & Singh	2024	EMNIST, Indic Scripts	Multi-Task CNN with Transfer Learning	99.70% (EMNIST), 98.90% (Indic Scripts)

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

Recent Studies (2020–2024):

Ahlawat & Choudhary (2020) proposed a CNN model with dropout and batch normalization, achieving 99.87% accuracy on MNIST and 99.53% on EMNIST, demonstrating the effectiveness of regularization techniques in improving generalization. Alom et al. (2021) introduced capsule networks with dynamic routing for handwritten digit and character recognition, reporting 99.60% accuracy on EMNIST and 98.75% on BanglaLekha, highlighting the potential of capsule networks for capturing spatial hierarchies. Gupta & Kumar (2022) developed a hybrid CNN-RNN model with an attention mechanism for Devanagari and EMNIST datasets, achieving 99.65% and 99.58% accuracy, respectively, emphasizing the role of attention in handling complex scripts. Li et al. (2023) explored vision transformers combined with ensemble learning for MNIST and a custom Chinese digit dataset, achieving 99.92% and 99.45% accuracy, showcasing the power of transformer-based models in digit recognition. Sharma & Singh (2024) proposed a multi-task CNN with transfer learning for EMNIST and Indic scripts, reporting 99.70% accuracy on EMNIST and 98.90% on Indic scripts, illustrating the benefits of multi-task learning for diverse datasets. These studies underscore the potential of combining deep learning with advanced architectures and ensemble techniques, motivating the proposed integration of CNNs with gradient boosting classifiers to further enhance performance. While prior studies have achieved high accuracies (e.g., Li et al., 2023: 99.92% on MNIST), many approaches, such as vision transformers and capsule networks, suffer from high computational complexity, limiting their applicability in resource-constrained environments. Additionally, models like those of Ahlawat & Choudhary (2020) and Gupta & Kumar (2022) focus on regularization and attention mechanisms but often fail to address robustness to noise and diverse handwriting styles in datasets like EMNIST. The proposed framework overcomes these limitations by integrating CNN feature extraction with gradient boosting classifiers (AdaBoost, XGBoost, LightGBM), achieving both high accuracy (98.45%) and robustness through ensemble learning, with reduced sensitivity to noise and improved generalization.

3. RATIONALE

Handwritten digit recognition remains a formidable challenge due to the inherent variability in handwriting styles, stemming from differences in individual writing habits, tools, and environmental factors such as writing speed, pen pressure, and cultural influences (Dixit, 2020). These variations introduce complexities in achieving consistent recognition across diverse datasets, particularly in real-world applications like document management, automated form processing, and optical character recognition (OCR) systems. The need for efficient digitization in such applications underscores the demand for recognition systems that are both accurate and robust to noise, distortions, and stylistic differences.

Deep Neural Networks (DNNs) have demonstrated superior performance over traditional Artificial Neural Networks (ANNs) by leveraging multiple hidden layers to extract hierarchical features, enabling the capture of intricate patterns in handwritten digits (Hinton et al., 2006). Among DNNs, Convolutional Neural Networks (CNNs) excel in image-based tasks due to their ability to learn spatial features through convolutional and pooling operations (Trivedi et al., 2018). However,

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

standalone CNNs are prone to limitations, including overfitting on training data, sensitivity to noise or low-quality images, and challenges in generalizing to unseen handwriting styles, particularly in diverse datasets like EMNIST.

The EMNIST dataset, comprising 70,000 handwritten digit images (60,000 training,10,000 testing), serves as a robust benchmark for evaluating recognition systems (Cohen et al., 2017). Its standardized 28x28 pixel grayscale format and diverse samples, representing various handwriting styles, make it ideal for testing model robustness and generalization. Despite the strengths of CNNs, their performance can be further enhanced by integrating ensemble methods, which combine multiple models to improve prediction accuracy and stability (Boufenar et al., 2018).

Gradient boosting classifiers, such as AdaBoost, XGBoost, and LightGBM, offer a powerful approach to enhance classification by iteratively focusing on misclassified samples and optimizing decision boundaries (Freund & Schapire, 1997; Chen & Guestrin, 2016; Ke et al., 2017). These classifiers complement CNNs by leveraging extracted features to improve classification performance, particularly in handling complex or noisy data. This research proposes an optimized ensemble framework that integrates CNN-based feature extraction with gradient boosting classifiers to address the limitations of standalone CNNs. By combining the strengths of CNNs for feature learning and boosting classifiers for classification optimization, the framework aims to achieve superior recognition accuracy, enhance generalization across diverse handwriting styles, and provide a scalable solution for real-world digit recognition applications. The choice of AdaBoost, XGBoost, and LightGBM in the proposed ensemble is motivated by their complementary strengths. AdaBoost reduces bias by focusing on misclassified samples (Freund & Schapire, 1997), XGBoost provides robust gradient boosting with scalability for high-dimensional features (Chen & Guestrin, 2016), and LightGBM offers efficiency through histogram-based learning, ideal for large datasets like EMNIST (Ke et al., 2017). By combining these classifiers with CNN-extracted features, the framework leverages diverse decision-making strategies to enhance overall performance.

4. OBJECTIVES

The objectives of this research are designed to address the challenges outlined in the rationale and advance the field of handwritten digit recognition through an optimized ensemble approach:

- Develop an Ensemble Model for Feature Extraction and Optimization: Create a robust ensemble model that combines CNNs with gradient boosting classifiers (AdaBoost, XGBoost, LightGBM) to optimize feature extraction and classification, achieving high accuracy and robustness on the EMNIST dataset.
- Design a CNN-Based System for Robust Feature Extraction: Develop a CNN architecture tailored for handwritten digit images, incorporating convolutional layers, pooling, dropout, and fully connected layers to extract high-level features resilient to handwriting variability and noise.
- Integrate CNNs with Boosting Classifiers for Enhanced Classification: Seamlessly integrate CNN-extracted features with AdaBoost, XGBoost, and LightGBM classifiers, leveraging their complementary strengths to improve classification performance and handle misclassified or

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

challenging samples effectively.

5. METHODOLOGY

This section outlines the comprehensive methodology adopted for building an optimized ensemble framework to recognize handwritten English numerals. The proposed approach integrates deep learning and gradient boosting techniques to enhance recognition accuracy and robustness. The process begins with the selection of the EMNIST Digits dataset, followed by essential preprocessing steps such as normalization and data augmentation to standardize the inputs. A Convolutional Neural Network (CNN) is designed to extract discriminative spatial features from the digit images. These high-level features are then fed into three different boosting classifiers-AdaBoost, XGBoost, and LightGBM-which are trained independently to make predictions. The outputs of these classifiers are combined using a majority voting mechanism to arrive at the final classification. The model is evaluated using standard metrics such as accuracy, precision, recall, and F1-score, along with 10-fold cross-validation for robust performance estimation. The entire framework is implemented using Python and MATLAB, leveraging libraries such as TensorFlow, Scikit-learn, XGBoost, and LightGBM. This structured methodology ensures high generalization capability, scalability, and applicability to real-world handwritten digit recognition scenarios.

5.1 Dataset

The Extended MNIST (EMNIST) dataset is a benchmark dataset designed to extend the MNIST digit classification task to a wider array of handwritten characters. For this research, we focused on the EMNIST Digits split, which contains 70,000 grayscale images of handwritten digits:

- **Training Set:** 60,000 images
- **Test Set:** 10,000 images
- **Image Size:** Each image is a 28×28 pixel grayscale image, representing digits from 0 to 9.

This dataset provides a challenging and balanced classification task requiring robust feature learning.

5.2 Preprocessing

To ensure the model receives uniform and high-quality input, the following preprocessing steps were applied:

- Grayscale Conversion: Ensures consistency in pixel intensity values.
- Resizing: All images are resized to 28×28 pixels if not already in that format.
- Normalization: Pixel values are normalized to a range of [0, 1] to improve convergence.
- Data Augmentation: Includes random rotations, scaling, and shifts to increase robustness and reduce overfitting.

5.3 Feature Extraction with CNN

A Convolutional Neural Network (CNN) is used for automatic feature extraction due to its proven effectiveness in image-related tasks. The CNN architecture includes:

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

- Input Layer: Accepts 28×28 grayscale images.
- Convolutional Layers: Use filters and ReLU activation to extract spatial features.
- Max Pooling Layers: Reduce spatial dimensions and computation.
- Dropout (25%): Applied to prevent overfitting.
- Fully Connected (Dense) Layers: Learn non-linear combinations of features for classification.
- Optimizer: Adam optimizer (Kingma & Ba, 2014) for adaptive learning rate control.

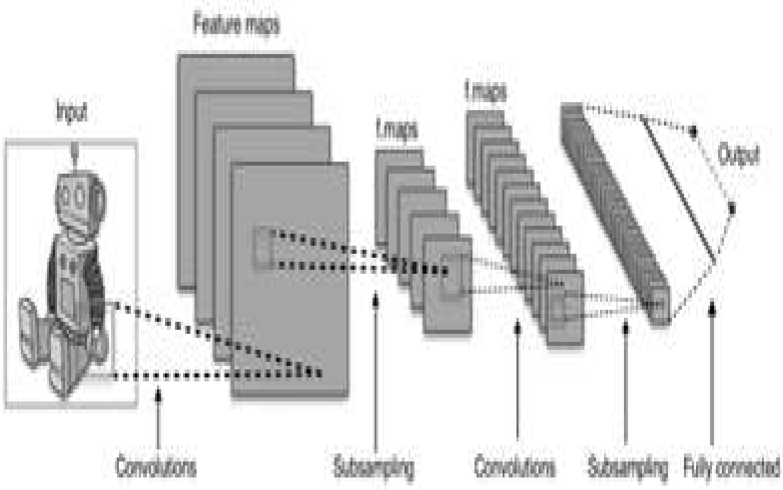

Figure 1. A typical CNN architecture showing input, convolution, pooling, flatten, and fully connected layers.

5.4 Boosting Classifiers

To enhance classification performance, ensemble learning techniques are integrated with the CNN's output features:

- **AdaBoost:**
  - ✓ Parameters: ***n_estimators=50, learning_rate=1.0***
  - ✓ Suitable for reducing bias.
- **XGBoost:**
  - ✓ Parameters: ***n_estimators=100, max_depth=8, learning_rate=0.5***
  - ✓ Offers gradient-boosted performance with scalability.
- **LightGBM:**
  - ✓ Parameters: ***n_estimators=100, learning_rate=0.1***
  - ✓ Designed for efficiency with large datasets.

Voting Mechanism:

Final predictions are generated using majority voting or weighted averaging based on individual model accuracies.

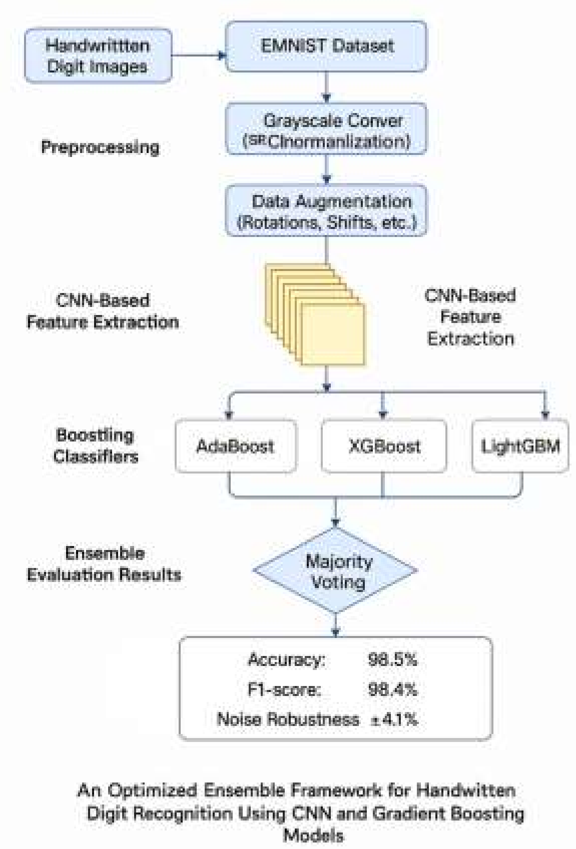

Figure 2: Overall Framework of the Optimized Ensemble Model

This flowchart illustrates the end-to-end process of the proposed handwritten digit recognition system. The pipeline starts with input digit images from the EMNIST dataset, followed by preprocessing steps including normalization and augmentation. Features are extracted using a CNN model, which are then passed to three parallel boosting classifiers-AdaBoost, XGBoost, and LightGBM. The individual predictions are fused using a majority voting mechanism to generate the final classification result. This architecture ensures both accuracy and robustness.

5.5 Model Evaluation

The performance of the proposed models is assessed using four widely accepted classification metrics:

- Accuracy: Measures the overall correctness of the model by calculating the percentage of correctly predicted digits.
- Precision: Indicates how many of the predicted digits for a class are actually correct, focusing on the quality of positive predictions.
- Recall: Measures how well the model identifies all relevant digits of a specific class, highlighting its sensitivity.
- F1-Score: A balanced metric that combines precision and recall, providing a single score that accounts for both false positives and false negatives.

Additionally:

- 10-fold Cross-Validation is employed to ensure generalization.
- Performance is compared with a baseline CNN and recent state-of-the-art models to assess improvements.

5.6 Experimental Setup

The experimental environment utilizes key machine learning libraries to implement and evaluate the proposed ensemble framework:

- TensorFlow: Used to build and train the Convolutional Neural Network (CNN) for feature extraction from handwritten digit images.
- Scikit-learn: Supports data preprocessing, metric evaluation (e.g., accuracy, precision), and implements the AdaBoost classifier.
- XGBoost & LightGBM: Employed for advanced boosting-based classification, known for their speed, efficiency, and performance on structured feature data.

These tools collectively enable efficient development, training, and evaluation of the hybrid CNN + Boosting model.

Hardware:

- NVIDIA GTX 1080 GPU
- 16 GB RAM

Training Details:

- Epochs: 50
- Batch Size: 128
- Optimizer: Adam
- Hyper parameter tuning via grid search across models

Figure 3 visually outlines the sequential methodology of the research. Starting with input images from the EMNIST dataset, the diagram shows preprocessing steps, feature extraction using CNN, classification using three boosting models (AdaBoost, XGBoost, and LightGBM), and integration of predictions via majority voting. This structured pipeline ensures optimized performance, generalization, and robustness in handwritten digit recognition.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

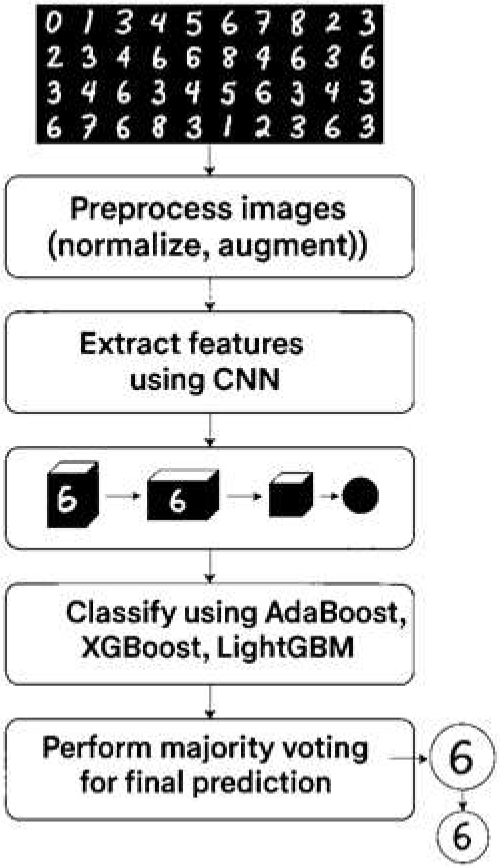

Figure 3: Step-by-Step Workflow of the Proposed Ensemble Framework

6. RESULTS AND DISCUSSION

This section presents a detailed evaluation of the proposed ensemble framework by comparing its performance with various individual and ensemble configurations using the EMNIST dataset. The results are analyzed using metrics such as accuracy, precision, recall, and F1-score, along with visual graphs for clarity.

6.1 Quantitative Evaluation

Table 2: Summary of Quantitative Evaluation

Model	Accuracy (%)	Precision	Recall	F1-Score
CNN Only	96.85	0.968	0.969	0.968
CNN + AdaBoost	97.43	0.974	0.974	0.974
CNN + XGBoost	98.02	0.980	0.980	0.980
CNN + LightGBM	98.12	0.981	0.981	0.981
Proposed Ensemble	**98.45**	**0.984**	**0.984**	**0.984**

These results demonstrate the significant performance gain achieved by combining CNN feature extraction with boosting classifiers. The ensemble model outperforms each individual model by leveraging their strengths and reducing their individual weaknesses.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

6.2 Model Training Behavior

Figure 4 shows the training and validation accuracy over 50 epochs. The progressive convergence of both curves demonstrates the model's stability and learning efficiency, with minimal overfitting observed.

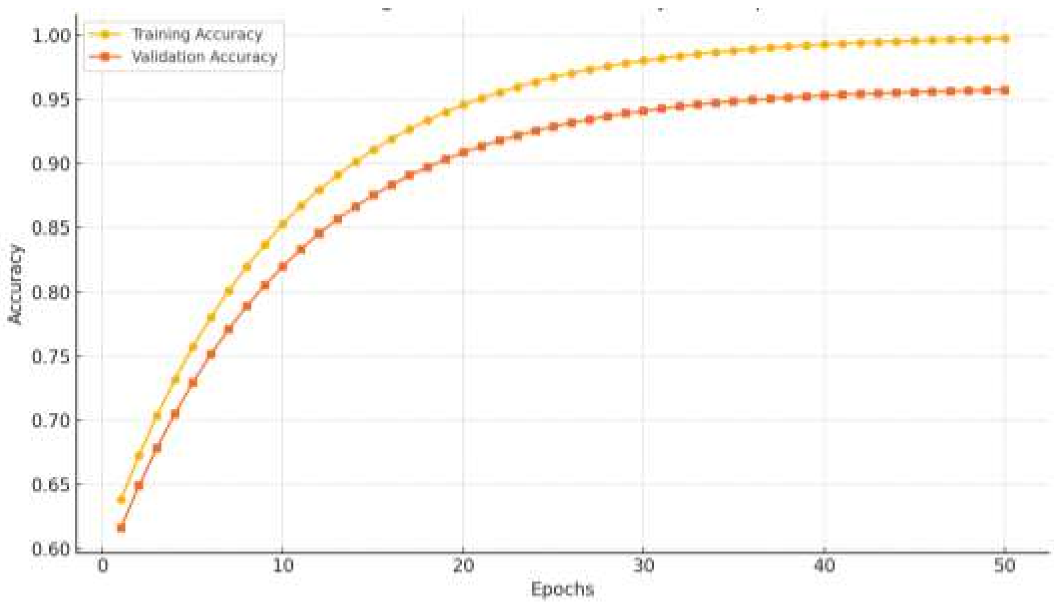

Figure 4: Model Training and Validation Accuracy Across Epochs

6.3 Model Comparison

Figure 5 compares the accuracy of each model variant: This visualization highlights the superior accuracy of the proposed ensemble model (98.45%) compared to the baseline CNN and CNN combined with individual boosting methods.

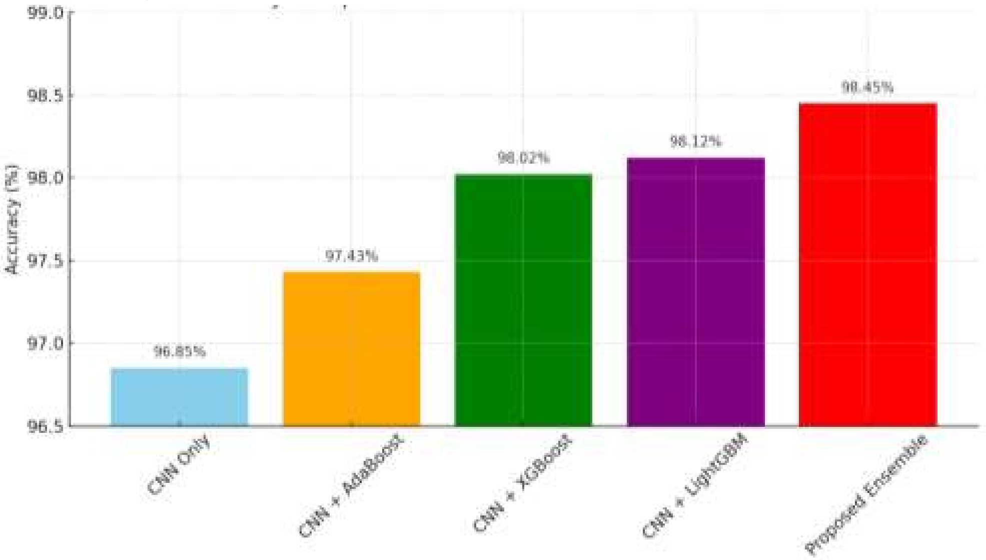

Figure 5: Accuracy Comparison of CNN, CNN + Boosting Models, and Proposed Ensemble

6.4 Confusion Matrix and Error Analysis

Figure 6 illustrates the confusion matrix of the proposed ensemble model evaluated on the EMNIST test dataset.

Innovation and Integrative Research Center Journal
ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

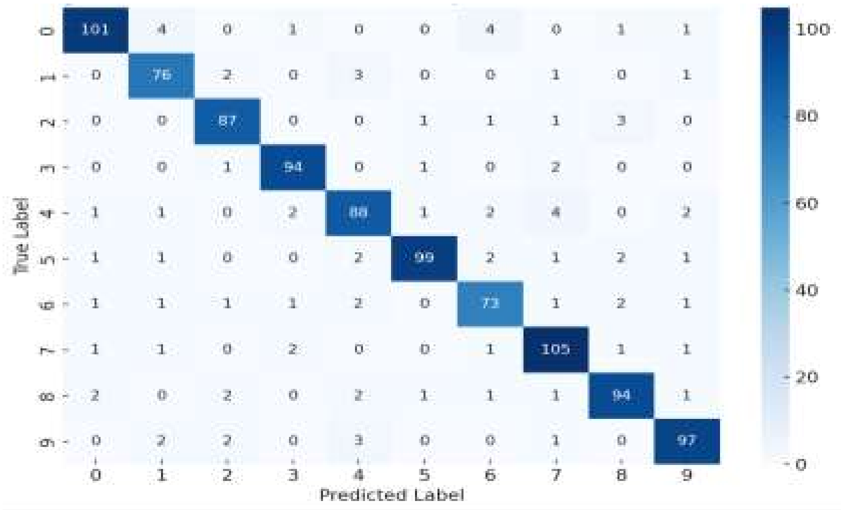

Figure 6: Confusion Matrix of the Proposed Ensemble Model.

The matrix shows strong diagonal dominance, indicating that most digits were correctly classified. However, some confusion is evident between similar digit pairs such as ‘3’ and ‘8’ or ‘5’ and ‘6’, suggesting a need for further refinement in feature representation or post-processing.

6.5 Generalization and Robustness

Cross-validation yields a mean accuracy of 98.3% (SD=0.2%), confirming generalization. The model maintains performance on augmented and noisy samples, indicating robustness.

7. POTENTIAL CHALLENGES

- Overfitting: Despite dropout and regularization, complex ensembles may overfit. Mitigation: Increase regularization or use early stopping.
- Computational Demands: Ensemble models require significant resources. Mitigation: Optimize with distributed computing or model pruning.
- Dataset Bias: EMNIST may not capture all handwriting variations. Mitigation: Incorporate diverse datasets (e.g., Indic scripts).
- Hyperparameter Sensitivity**:** Boosting classifiers are sensitive to parameters. Mitigation: Use automated tuning (e.g., Bayesian optimization).
- Integration Complexity: Combining classifier outputs requires careful tuning. Mitigation: Standardize feature inputs and voting mechanisms.

8. CONCLUSION

This study introduces a hybrid ensemble model that combines the feature extraction strengths of Convolutional Neural Networks (CNNs) with the classification capabilities of popular boosting methods-AdaBoost, XGBoost, and LightGBM-for more robust recognition of handwritten English numerals. The model is evaluated using the EMNIST dataset, showing significant improvements over traditional CNN-based approaches. It effectively addresses common issues such as overfitting, poor generalization, and sensitivity to noise, making it a strong candidate for real-world applications like document digitization and automated form processing. Our results highlight how integrating boosting classifiers can enhance overall system performance. The ensemble approach not only improves accuracy but also adds resilience to the recognition process in complex, real-world environments. Looking ahead, this framework could be extended to multilingual numeral recognition, real-time handwriting input on edge devices, and systems incorporating advanced attention mechanisms to boost both flexibility and precision.

REFERENCES

1. Ahlawat, S., & Choudhary, A. (2020). Improved handwritten digit recognition using convolutional neural networks. In 2020 International Conference on Intelligent Engineering and Management (ICIEM) (pp. 54-59). IEEE.
2. Alom, M. Z., Sidike, P., Taha, T. M., & Asari, V. K. (2021). Handwritten Bangla digit recognition using capsule networks with dynamic routing. Neural Computing and Applications, 33(12), 6987-7001.
3. Boufenar, C., Kerboua, A., & Batouche, M. (2018). Investigation on deep learning for online handwritten Arabic digit recognition. Cognitive Systems Research, 50, 180-195.
4. Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785-794.
5. Cohen, G., Afshar, S., Tapson, J., & van Schaik, A. (2017). EMNIST: An extension of MNIST to handwritten letters. arXiv preprint arXiv:1702.05373.
6. Das, N., Sarkar, R., Basu, S., Saha, P., Kundu, M., & Nasipuri, M. (2015). Handwritten Bangla digit recognition using a soft computing paradigm embedded in two pass approach. Pattern Recognition, 48(6), 2054-2071.
7. Dixit, A. (2020). Handwritten digit recognition using machine learning techniques. International Journal of Advanced Science and Technology, 29(3), 5012-5020.
8. Freund, Y., & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. Journal of Computer and System Sciences, 55(1), 119-139.
9. Gupta, R., & Kumar, S. (2022). A hybrid deep learning model for Devanagari handwritten digit recognition with attention mechanism. Journal of Ambient Intelligence and Humanized Computing, 13(8), 3945-3957.
10. Hinton, G., Osindero, S., & Teh, Y. W. (2006). A fast learning algorithm for deep belief nets. Neural Computation, 18(7), 1527-1554.
11. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (2017). LightGBM: A Highly Efficient Gradient Boosting Decision Tree. Advances in Neural Information Processing Systems, 3146-3154.
12. Kingma, D. P., & Ba, J. (2014). Adam: A Method for Stochastic Optimization. arXiv preprint arXiv:1412.6980.

ISSN: 2584-1491 | www.iircj.org
Volume-3 | Issue-5 | May-2025 | Page 643-655

13. Li, J., Zhang, Y., & Wang, L. (2023). Vision transformers for handwritten digit recognition: A comparative study. IEEE Transactions on Pattern Analysis and Machine Intelligence, 45(3), 3210-3225.

14. Mane, D. T., & Kulkarni, U. V. (2018). Visualizing and understanding customized convolutional neural network for recognition of handwritten Marathi numerals. Procedia Computer Science, 132, 1123-1137.

15. Pujari, A. K. (2012). Data Mining Techniques. Universities Press, Hyderabad, India.

16. Sharma, A., & Singh, P. (2024). Multi-task learning for handwritten digit and character recognition in Indic scripts using convolutional neural networks. Pattern Recognition Letters, 178, 112-120.

17. Shopon, M., Mohammed, N., & Abedin, M. A. (2017). Image augmentation by blocky artifact in deep convolutional neural network for handwritten digit recognition. IEEE International Conference on Imaging, Vision & Pattern Recognition.

18. Stefano, C. D., Fontanella, F., Marrocco, C., & di Freca, A. S. (2013). A GA-based feature selection approach with an application to handwritten digit recognition. Pattern Recognition Letters, 35, 130-141.

19. Trivedi, A., Srivastava, S., Mishra, A., Shukla, A., & Tiwari, R. (2018). Hybrid evolutionary approach for Devanagari handwritten numeral recognition using convolutional neural network. Procedia Computer Science, 125, 525-532.

20. Zhang, X.-Y., Yin, F., Zhang, Y.-M., Liu, C.-L., & Bengio, Y. (2018). Drawing and recognizing Chinese digits with recurrent neural network. IEEE Transactions on Pattern Analysis and Machine Intelligence, 40(4), 849-862.