FusionNet: A KNN-MLP Hybrid Model for Bengali Handwritten Digit Recognition using HOG and LBP Features

Anower Hossen*, Muhammad Aman Ullah
Department of Computer Science & Engineering, International Islamic University Chittagong, Chittagong, Bangladesh
E-mail: a.h.sumon2607@gmail.com, aman_cse@iiuc.ac.bd
*Corresponding author

Keywords: FusionNet, K-Nearest Neighbor, NumtaDb, EfficientNet-B0

Received: May 4, 2025

Recent years have seen a surge of interest in research related to Bengali handwritten digit recognition, largely driven by its significant practical relevance and the pervasive utilization of the Bengali language. Convolutional Neural Networks (CNNs) have demonstrated notable success in this domain; however, hybrid approaches that integrate handcrafted feature extraction with conventional machine learning classifiers are emerging as effective alternatives. This study proposes and evaluates FusionNet, a hybrid model that combines the strengths of feature-based and learning-based methods through a two-stage classification pipeline. First, an optimized K-Nearest Neighbors (KNN) classifier generates a coarse label prediction based on handcrafted features. This prediction is then incorporated with origainal feature then fed into a Multi-Layer Perceptron (MLP), which performs the final classification. To enhance the system's robustness and generalization, few preprocessing techniques such as, binarization, Otsu's threshold, and data augmentation were implemented. Then, two complementary feature extraction techniques were applied. Firstly, Histogram of Oriented Gradients (HOG) is utilized; and secondly, Local Binary Patterns (LBP). These features were computed in parallel to mitigate runtime overhead, thereby enabling reduced runtime. FusionNet's performance was benchmarked against EfficientNet-B0, a state-of-the-art pre-trained CNN model, using two datasets: a custom dataset reflecting diverse handwriting styles and the publicly available NumtaDb dataset. FusionNet attained an accuracy of 87% on the custom dataset and 96% on NumtaDb. In comparison, EfficientNet-B0 achieved 91% and 97%, respectively. Although EfficientNet-B0 exhibited marginally superior accuracy, FusionNet exhibited superior efficiency and lower computational demands, thus rendering it a compelling candidate for deployment in resource-constrained environments.

Povzetek: Opisan je hibridni model FusionNet za prepoznavanje bengalskih ročno pisanih številk, ki združuje metodo KNN in večplastno nevronsko mrežo (MLP) z značilkami HOG in LBP. Predlagani pristop izboljša točnost prepoznave ter dosega večjo robustnost v primerjavi s posameznimi klasifikacijskimi modeli.

1 Introduction

The accurate recognition of handwritten digits constitutes a fundamental problem in the field of optical character recognition (OCR) and computer vision. This problem has significant implications for various real-world applications, including automated data entry, postal code sorting, and document digitization. Significant progress has been made in the field of digit recognition for Latin scripts, as evidenced by the high performance on datasets such as MNIST. However, the recognition of digits in scripts with more intricate structures, such as Bengali, poses unique and persistent challenges. Bengali, a language that is spoken by a significant number of people worldwide, possesses a rich and complex script. While the numerals are distinct, they often exhibit subtle shape similarities, even in their printed forms. This can complicate automated recognition [7]. The inherent variability introduced by individual handwriting styles, including differences in stroke thickness, slant, size, and overall form—further exacerbates this challenge. Consequently, robust recognition of handwritten Bengali characters is a critical problem with numerous practical applications, including general handwritten character recognition (HCR), optical character recognition (OCR) systems for documents, and word recognition [6]. The majority of models proposed for Bengali digit recognition have historically been rooted in CNN based pattern recognition and machine learning techniques [8]. While these approaches have laid the foundation for future progress, the increasing demand for higher accuracy, robustness, and adaptability across diverse writing styles necessitates the exploration of more advanced and resilient methodologies. A pivotal element of this intelligence pertains to the capacity of computers to accurately comprehend and identify alphabets and numerals across diverse languages spoken by humans. The recognition of numerals has emerged as a highly

active area of research in AI due to the inherent complexities it presents [9]. Given Bengali's global prominence and its integration into intelligent systems and machines, where numeral recognition is often crucial, its integration into such systems and machines is increasingly imperative. Despite the mounting interest, the extant body of work specifically addressing Bengali handwritten digit recognition, particularly using advanced neural network architectures, remains relatively limited. There is considerable potential for enhancement, particularly with regard to model robustness against varied handwriting, the management of imbalanced datasets, the assurance of flexibility across diverse writing styles, and the attainment of enhanced generalization capabilities across different datasets [10].

The objective of this study is to address the aforementioned gaps by introducing FusionNet, a novel hybrid model for Bengali handwritten digit recognition. FusionNet diverges from the prevailing trend of purely Convolutional Neural Network (CNN)-based approaches by integrating a K-Nearest Neighbors (KNN) classifier with a Multi-Layer Perceptron (MLP) within a two-stage framework. The efficacy of FusionNet is rigorously evaluated and compared against EfficientNet-B0, a state-of-the-art pre-trained deep learning model. The evaluation process employs a bespoke dataset, meticulously crafted to encompass a comprehensive spectrum of handwriting variations, in conjunction with NumtaDb, a preeminent benchmark dataset for Bengali digits. The comparative analysis between our custom dataset and the benchmark dataset provides empirical justification for FusionNet's performance and generalization capabilities. Furthermore, to enhance feature extraction efficiency and mitigate computational overhead, parallel processing techniques are strategically employed within FusionNet's architecture. This research makes a contribution to the field by developing an efficient and robust system for Bengali handwritten digit recognition. The proposed approach presents a compelling alternative to computationally intensive deep learning models, offering a novel solution to the challenges posed by traditional methods. The primary hypothesis tested in this study is: Hybrid models that integrate both traditional and deep learning components (e.g., KNN and MLP) can outperform or match the performance of conventional CNN-based models on low-resource or noisy handwritten Bengali digit datasets while reducing computational complexity.

- Combines traditional machine learning (KNN) with deep learning (MLP),
- Fuses features extracted from two distinct sources: handcrafted features (HOG, LBP), and outputs from KNN, then classify them via a lightweight neural network.
- Reduces dependency on purely deep architectures and introduces a parallelized pipeline for computational efficiency.

The document is organized in the following manner: Section 2 examines previous research on Bengali handwritten digit recognition, focusing on traditional methods, deep learning techniques, and hybrid approaches. Section 3 elaborates on the proposed methodology, detailing aspects such as preprocessing, feature extraction, model architecture, and training processes. Section 4 showcases the experimental outcomes, comparative assessments, and evaluations based on visualization. Section 5 discusses the results, emphasizing performance trade-offs, constraints, and potential enhancements. Lastly, Section 6 wraps up the study and suggests avenues for future exploration.

2 Literature review

The landscape of handwritten digit recognition has undergone continuous evolution, with early efforts predominantly reliant on CNN based pattern recognition and classification techniques. In the context of Bengali digits, analogous methodologies were initially predominant. Moreover, recent years have witnessed a significant surge in the application of more robust and sophisticated models, particularly those based on deep learning and innovative hybrid architectures, which have achieved remarkable accuracies across various scripts. The advent of deep learning, particularly Convolutional Neural Networks (CNNs), has led to a paradigm shift within the field, resulting in state-of-the-art performance. In the context of Bengali handwritten digit recognition, recent studies have employed sophisticated techniques, including: Dalui et al. (2024) employed a deep convolutional neural network on the unprocessed and extensively augmented NumtaDb dataset, attaining remarkable accuracies of 99% on the training set and 98% on the validation set [1]. Researchers have also focused on enhancing CNN architectures. Azgar et al. (2024) proposed a Dual-Input Convolutional Neural Network (DICNN) by modifying a standard Convolutional Neural Network (CNN) for the recognition of MNIST digits [2]. Hybrid models, which integrate elements from both traditional and deep learning paradigms, have also demonstrated considerable potential for Bengali digit recognition. In scenarios characterized by a paucity of data. Ahamed et al. (2024) introduced the SynergiProtoNet model, which employs few-shot learning on the NumtaDb dataset. Their research yielded encouraging results for languages or datasets with limited resources and samples, achieving accuracies of 90% for monolingual intra- datasets, 81% for monolingual inter-datasets, and 82% for cross-lingual datasets [18]. Khudeyer and Moosawi modified last layer of ResNet50 with Random Forest and Support Vector Machine, the result showed an increase in performance for Arabic Handwritten Character Dataset (AHCD), Alexa Isolated Alphabet Dataset (AIA9K), and Hijja Dataset [5]. Zhang et al. proposed a Chinese Medical Named Entity Recognition (MNER) method leveraging pre-trained models (RoBERTa, Word2Vec) and the efficient global pointer (EGP) which incorporated data augmentation, character-word fusion and improved decoding layer based on EGP [23]. Despite the substantial advancements and the high accuracies reported by numerous robust deep learning and hybrid techniques, critical challenges persist in Bengali handwritten digit recognition. These include

the high computational cost associated with training and deploying very deep networks, the need for models that exhibit greater flexibility and robustness with diverse and unconstrained real-world writing styles, improved generalization across various heterogeneous datasets, and enhanced performance on potentially imbalanced datasets [10]. Given that Bengali is one of the most widely spoken languages globally, its effective integration into intelligent machines and systems is imperative, particularly in the context of accurate numeral recognition, which can become a crucial component. This study presents FusionNet, a novel hybrid KNN-MLP model, which has been developed to address some of the aforementioned challenges by offering a computationally efficient yet highly accurate alternative to purely deep learning approaches. A comparison was made between FusionNet and EfficientNet-B0, a leading deep learning model, using both a custom-created dataset and the benchmark NumtaDb. This comparison provides empirical. justification for the performance and efficiency of FusionNet, particularly through the strategic employment of parallel processing for feature extraction. As demonstrated in the following table 1, the range of research conducted is illustrated.

Table 1: Summary of related works

Author and year	Title	Methods/Algorithms	Findings
A. Dalui, R. Sarkar, S. Sharma, A. Ghosh et al. (2024)	A Deep Convolutional Neural Network Approach to Recognize Bangla Handwritten Digits	Deep Convolutional Neural Network	Achieved an accuracy of 99.6% on the training set and 98.65% on the validation set.
Ali, A., Senan, N., Murli, N. (2024).	Convolutional Neural Network Using Regularized Conditional Entropy Loss (CNNRCoE) for MNIST Handwritten Digits Classification.	Convolutional Neural Network using Regularized Conditional Entropy Loss (CNNRCoE)	Achieving an accuracy at about 98%.
Ahamed, M., Kabir, R.B., Dipto, T.T., Al Mushabbir, M., Ahmed, S. and Kabir, M.H. (2024)	Performance Analysis of Few-Shot Learning Approaches for Bangla Handwritten Character and Digit Recognition.	Few-shot learning: SynergiProtoNet	The model reliably attains high results in Monolingual Intra-Dataset, Monolingual Inter-Dataset, Cross-Lingual Transfer, and Split Digit assessments.
Ali Azgar, Nazir, Afsana, Hossain, Anwar et. al (2024)	MNIST Handwritten Digit Recognition Using a Deep Learning-Enhanced Dual Input Convolutional Neural Network (DICNN) Model	Deep learning-based modified Dual-Input CNN (DICNN)	accuracy and F1-score of the model are 98.9%, 99.9% and recall and precision is 99.7%, 99.3%.
Amin, Reza et.el. (2023)	A Fine-Tuned Hybrid Stacked CNN to Improve Bengali Handwritten Digit Recognition	LBP, CLBP, HOG, PCA. XGBoost classifier, three stacked CNN	XGBoost classifier achieved an accuracy of 85.29%, Stacked CNN reached 99.66% training accuracy and a 97.57% testing accuracy.
Sufian, A., Ghosh, A., Naskar (2022)	BDNet: Bengali Handwritten Numeral Digit Recognition based on Densely connected CNN	Densely connected deep convolutional neural network: BdNet	The model achieved a test accuracy of 99.775%. The BDNet model gives 62.5% error reduction compared to previous state-of-the-art models.
Maity, S., Dey et. al. (2020)	Handwritten Bengali character recognition using deep convolutional neural network.	Segmentation-based handwritten word recognition with neural network	Able to extract characters with 65% accuracy. Recognize the properly segmented alphabets with 99.5% accuracy.

Shawon, Rahman et. al. (2018)	Bangla Handwritten Digit Recognition Using Deep CNN for Large and Unbiased Dataset	Deep CNN with different kinds of preprocessing techniques	Deep CNN achieved 92.72% testing accuracy
Khudeyer, Moosawi (2023)	Combination of Machine Learning Algorithms and Resnet50 for Arabic Handwritten Classification	Modified last layer of ResNet50 with Random Forest and Support Vector Machine	Modified ResNet50 architecture has achieved a rate of 92.37%, 98.39%, and 91.64%, while the combination architecture has achieved 95%, 99%, and 92.4% for AIA9K, AHCD, and Hijja datasets
Zhang, Li et. El. (2025)	Chinese Medical Named Entity Recognition Using Pre-Trained Language Models and an Efficient Global Pointer Mechanism	Chinese MNER method using pre-trained models (RoBERTa, Word2Vec) and the Efficient Global Pointer	Achieves F1 scores of 75.87% and 92.77% on the CMeEE-V2 and CCKS2020 datasets. Outperforming the RoBERTa-BiLSTM-CRF baseline by 3.06% and 4.38%, respectively.

3 Methodology

This research utilizes a structured experimental framework to create and assess a hybrid model for recognizing handwritten Bengali digits, named FusionNet, comparing it to the leading deep learning benchmark, EfficientNet-B0. The approach consists of (1) collecting and preparing two varied datasets, one being a custom dataset and the other the NumtaDb dataset (2) implementing systematic preprocessing and augmentation to improve data quality, (3) extracting handcrafted features to identify patterns, and (4) training and assessing both the proposed and comparative models using standardized performance metrics.

The optimized MLP features three fully connected layers utilizing SELU activation, each accompanied by Batch Normalization and Dropout (excluding the final hidden layer). The output layer consists of 10 units with a Softmax activation for classification purposes. Hyperparameters including the number of units, dropout rates, and learning rate were fine-tuned with the help of Optuna.

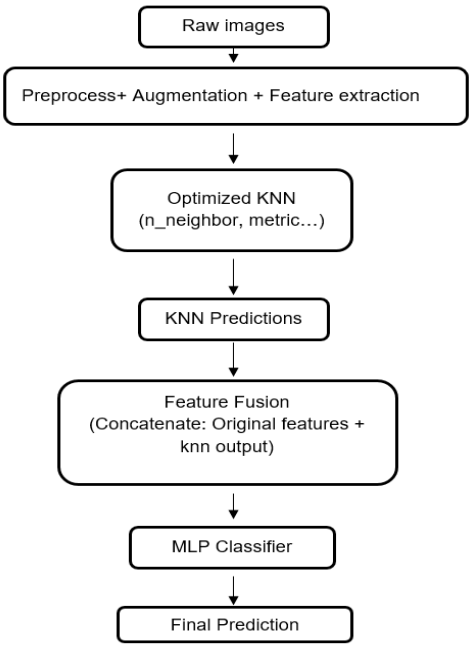

Figure 1: Overview of the working process.

3.1 Data collection

Two datasets were utilized to assess the performance and generalization of FusionNet. The custom dataset comprises 2,090 images that represent a variety of handwriting styles, featuring distortions, inconsistencies, and visually demanding qualities to reflect real-world variations in Bengali digits. The NumtaDb dataset, which is publicly accessible on Kaggle, served as a benchmark. It is structured similarly to MNIST but contains higher levels of noise and variability, making it ideal for evaluating the robustness of models as well as the effectiveness of preprocessing and hybrid classification techniques.

3.2 Data pre-processing

The input images were initially converted to grayscale to reduce computational complexity while preserving essential structural information. Thereafter, the images were resized to 28×28 pixels, a standard that aligns with prevailing conventions in the field of handwritten digit recognition. Otsu's thresholding binarized the images, clearly separating digits from the background. Finally, the pixel intensity values were normalized to the range [0, 1], thereby enhancing the efficiency of the training process and promoting numerical stability during the optimization of the model.

3.3 Data augmentation

To improve the diversity and strength of the training dataset, several data augmentation methods such as rotation, scaling, and shifting were utilized. These transformations mimicked real-world variations of handwritten digits, substantially enlarging the dataset size without the need for extra data collection. Data augmentation played a crucial role in enhancing the model's generalization, minimizing overfitting, and boosting performance, especially in cases with limited data. In addition, it bolstered the model's adaptability to noise and slight variations in input, thus increasing overall accuracy and dependability in recognizing handwritten digits.

3.4 Feature extraction

To achieve effective digit recognition, we utilized a combined approach involving Histogram of Oriented Gradients (HOG), Gabor Filters, and Local Binary Patterns (LBP), which allows us to capture different aspects of image information. HOG focuses on extracting structural patterns and edge orientations, Gabor filters are designed to capture textures and directional information, and LBP detects fine local texture patterns. When these techniques are used in combination, they produce a comprehensive and varied feature set that improves the model's capacity to generalize and accurately identify handwritten digits. We tested multiple feature combinations HOG + Gabor, Gabor + LBP, HOG + LBP, and all three together using the same preprocessing and classification pipeline. HOG + LBP produced the best accuracy, so it was adopted for the final FusionNet model.

3.5 Parallel processing

To address the substantial computational expense of the coding pipeline, we utilized CPU-based parallel computation to enhance efficiency. By leveraging the joblib library (Parallel, delayed) in conjunction with Python's multiprocessing module, we allocated tasks across the two available CPU cores, facilitating simultaneous execution.

3.6 Standardization

Since different techniques yield features with varying numerical ranges, all features were standardized to guarantee equal influence on model training.

$$\text{Xscaled} = \frac{X-\mu}{\sigma} \quad (1)$$

The original feature value (x) was adjusted using the mean (μ) and standard deviation (σ) derived from the training set, which were then applied to the test set. This approach prevents features with larger ranges from overshadowing others and ensures uniformity between training and testing.

3.7 FusionNet (Hybrid KNN-MLP)

FusionNet is a hybrid model that utilizes a two-stage approach, integrating K-Nearest Neighbors (KNN) and Multi-Layer Perceptron (MLP) for recognizing Bengali numerals. Initially, KNN conducts a preliminary classification by utilizing handcrafted features such as HOG and LBP, which capture local feature similarities. The predictions from KNN are then incorporated as an additional feature into the original feature set, which is subsequently processed by the MLP. The MLP, which includes dropout for regularization, is designed to model complex non-linear relationships to arrive at the final decision. This feature-level fusion capitalizes on KNN's ability to recognize local patterns and MLP's capacity for learning, resulting in enhanced accuracy and robustness.

3.8 EfficientNet-B0

EfficientNet-B0 serves as our state-of-the-art comparison model. It is a lightweight yet powerful deep learning model that achieves a balance between high accuracy and computational efficiency through a compound scaling approach, which uniformly scales network depth, width, and resolution. EfficientNet-B0, pre-trained on the large-scale ImageNet dataset [20], outperforms many traditional CNNs like ResNet and VGG while utilizing significantly fewer parameters. Its pre-trained features transfer effectively to the Bengali handwritten digit recognition task (NumtaDb), ensuring strong performance even without requiring excessively large amounts of training data specific to Bengali digits. Previous studies have consistently demonstrated EfficientNet's superior performance in various image classification tasks, reinforcing its reliability as a robust benchmark in this study.

3.8 EfficientNet-B0

EfficientNet-B0 serves as our state-of-the-art comparison model. It is a lightweight yet powerful deep learning model that achieves a balance between high accuracy and computational efficiency through a compound scaling approach, which uniformly scales network depth, width, and resolution. EfficientNet-B0, pre-trained on the large-scale ImageNet dataset [20], outperforms many traditional CNNs like ResNet and VGG while utilizing significantly fewer parameters. Its pre-trained features transfer effectively to the Bengali handwritten digit recognition task (NumtaDb), ensuring strong performance even without requiring excessively large amounts of training data specific to Bengali digits. Previous studies have consistently demonstrated EfficientNet's superior performance in various image classification tasks, reinforcing its reliability as a robust benchmark in this study.

3.9 Model training

The training process for both FusionNet and EfficientNet-B0 was meticulously controlled to ensure fair comparison and optimal performance. Both the model is trained for 20 epochs.

3.9.1 Data splitting

The following table illustrates data split ratio for both datas

Table 2: Dataset split ratio

Dataset	Total Image	Training	Testing	Split Ratio	Stratified
Custom Dataset (Digit: 0-9)	2090	1672	418	80% / 20%	True
NumtaDb (Digit: 0-9)	72,045	57,636	14,409	80% / 20%	True

3.9.2 Hyperparameter optimization

Hyperparameter optimization techniques were employed to identify the ideal set of parameters for each algorithm, which control how the models learn. Hyperparameters for both KNN and MLP models were optimized using Bayesian optimization via the Optuna framework. For the KNN model, performance was evaluated during tuning using stratified 5-fold cross-validation, implemented via cross_val_score. For the MLP model, tuning was performed using an 80/20 hold-out validation split, with validation accuracy guiding the Optuna search. After selecting the best architecture and hyperparameters, the final model was evaluated using stratified 5-fold cross-validation, reporting fold-wise accuracy.

Table 3: Best parameters for FusionNet

Algorithm	**Parameters (For Custom dataset)**
KNN	'n_neighbors': 3, 'metric': 'minkowski', 'weights': 'distance'
MLP	'num_units_1': 512, 'num_units_2': 192, 'num_units_3': 64, 'dropout_1': 0.322968016, 'dropout_2': 0.224990353, ‘learning_rate': 0.000732571
Algorithm	**Parameters (For NumtaDb dataset)**
KNN	'n_neighbors': 11, 'metric': 'manhattan', 'weights': 'distance'
MLP	'num_units_1': 448, 'num_units_2': 192, 'num_units_3': 96, 'dropout_1': 0.206206845, 'dropout_2': 0.269743632, 'learning_rate': 0.000981219

3.10 Performance evaluation metrics

Analytical techniques and conventional classification criteria were used to assess the models' performance**.** Accuracy measured overall correctness, whereas precision, recall, and F1-score offered assessments of predicted dependability and balance on a per-class basis. The misclassifications between the digits were examined using a confusion matrix. In order to assess feature separability even more, t-SNE visualizations were made. Together with AUC, precision-recall and ROC curves provided information about performance based on thresholds and discriminative skills. Finally, FusionNet and EfficientNet-B0 were statistically compared using the McNemar Test to see if the performance differences were statistically significant.

4 Results

This section presents a detailed analysis of FusionNet's classification performance on both the primary dataset and the benchmark NumtaDb dataset. We also provide a comparative evaluation against EfficientNet-B0, a state-of-the-art deep learning model, and discuss the implications of our findings, including visualization of the learned feature spaces.

4.1 FusionNet performance on primary dataset

The cross-validation score and classification results for FusionNet on the custom primary dataset are summarized in Table 4 & 5. The model demonstrated impressive overall performance by utilizing various feature combinations. When employing HOG and LBP, it achieved an accuracy of 87% and a macro-average precision of 88.25%. The combination of Gabor with LBP resulted in 82% accuracy, while Gabor paired with HOG produced an accuracy of 81%. When all three features were combined, the model attained an accuracy of 83%. Based on these findings, we determined that HOG and LBP represented the optimal combination, dropping Gabor.

Table 4: Cross-validation summary

Per-fold Accuracies:	0.8746, 0.8627, 0.9042, 0.8683, 0.8593
Mean Accuracy	0.8738
Standard Deviation	0.0161

Table 5: Classification report of FusionNet on primary dataset

Class	Precision	Recall	F1-Score	Support
0	0.9474	0.9250	0.9361	40
1	0.7907	0.7750	0.7828	40

2	0.9130	0.8750	0.8936	40
3	0.8721	0.9000	0.8858	40
4	0.9250	0.9048	0.9148	42
5	0.8846	0.8571	0.8706	42
6	0.8421	0.8000	0.8205	40
7	0.9565	0.9778	0.9670	45
8	0.9318	0.9070	0.9192	43
9	0.7619	0.7000	0.7298	46
Accuracy			0.8703	418
Macro Avg	0.8825	0.8629	0.8720	418
Weighted Avg	0.8765	0.8703	0.8721	418

Cross-validation reveals an average accuracy of 87.38%, exhibiting moderate variability among folds (standard deviation 0.0161), which suggests a generally stable yet somewhat fluctuating performance. Class-wise evaluation revealed consistently high recognition rates for digits 0, 4, 7, and 8, each exceeding 90% in both precision and recall. Digits 2, 3, 4, 5, and 6 also demonstrated strong performance, maintaining F1-scores around 88%. In contrast, digits 1 and 9 exhibited comparatively lower recognition, with F1-scores of 78.28% and 72.98%, respectively, indicating higher misclassification in these categories. The overall accuracy of 87% across all ten classes highlights the model's ability to maintain balanced performance, while also identifying specific digits that require further refinement. These results establish FusionNet as a reliable framework for Bengali numeral recognition, offering a strong foundation for integration into practical OCR systems.

4.2 FusionNet performance on numtadb dataset

The overall accuracy achieved on NumtaDb is 96%, with macro and weighted averages for precision, recall, and F1-score also around 96.3%, indicating balanced performance across all classes.

Table 6: Cross-validation summary

Per-fold Accuracies	0.9989, 0.9984, 0.9980, 0.9990, 0.9991
Mean Accuracy	0.9987
Standard Deviation	0.0004

Table 7: Classification report of FusionNet on NumtaDb

Class	Precision	Recall	F1-Score	Support
0	0.9750	0.9696	0.9723	1448
1	0.9260	0.9562	0.9409	1439
2	0.9839	0.9730	0.9784	1445
3	0.9622	0.9509	0.9566	1447
4	0.9541	0.9738	0.9638	1450
5	0.9585	0.9565	0.9575	1448
6	0.9548	0.9521	0.9535	1442
7	0.9739	0.9875	0.9806	1436
8	0.9901	0.9832	0.9867	1431
9	0.9501	0.9241	0.9369	1423
Accuracy			0.9627	14409
Macro Avg	0.9629	0.9627	0.9627	14409
Weighted Avg	0.9629	0.9627	0.9627	14409

Class-wise evaluation shows exceptionally high recognition rates for most digits, with precision and recall exceeding 97% for digits 0, 2, 7, and 8. Digits 3, 4, 5, and 6 maintained strong F1-scores around 95–96%. Slightly lower performance was observed for digits 1 and 9, which recorded F1-scores of approximately 94% and 93%, respectively. These results reflect a robust and well-generalized classifier with only minor variations among specific digits. Cross-validation confirms the consistency, demonstrating a mean accuracy of 99.87% with minimal fluctuations among the folds. And, there is no significance difference between training and testing accuracy which suggests no overfitting occurred.

4.3 Confusion matrix analysis

The confusion matrix for custom dataset demonstrates strong diagonal dominance, confirming that most digits are correctly classified.

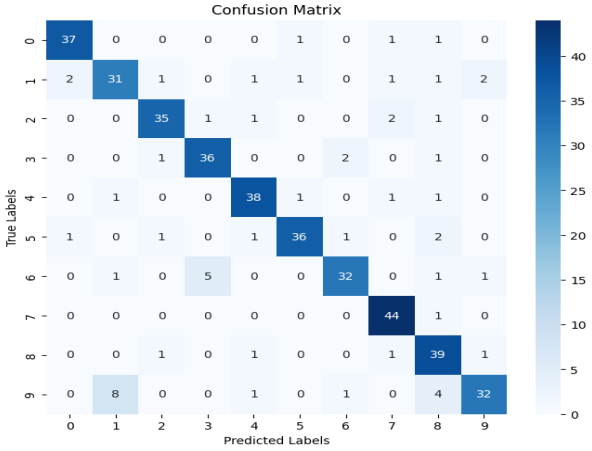

Figure 2: Confusion matrix of FusionNet on primary

Digits 4, 7, and 8, achieved the highest correct predictions with minimal misclassifications. Moderate confusion is observed between certain digits, such as 9 misclassified as 1,8 (8 and 4 cases) and 6 misclassified as 3 (5 cases). Digits 1, 2 and 5 also show occasional misclassifications. Overall, the matrix reflects consistent and balanced performance, with only minor overlaps between visually similar digits. The results indicate that the classifier

maintains high reliability even in challenging scenarios involving ambiguous digit shapes.

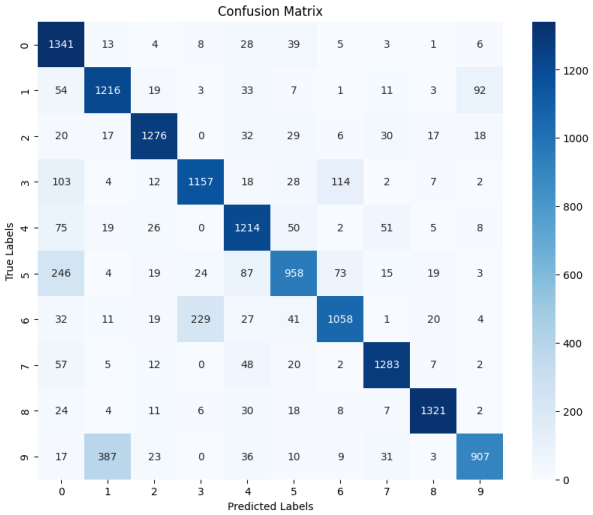

Figure 3: Confusion matrix of FusionNet on NumtaDb

The confusion matrix for NumtaDb also indicates accurate classification across most classes. Digit 0 achieved the highest correct predictions (1341), followed closely by digits 8 (1321), digits 7 (1283) and 2 (1276). Misclassifications are primarily concentrated among visually similar digits, with the most notable cases being 9 predicted as 1 (387 instances), 5 predicted as 0 (246 instances) and 6 predicted as 3 (229 instances). Additional errors are observed between digits 1,2,3 and 5, albeit at lower frequencies. Digits 7 and 8 demonstrate minimal confusion with other classes, suggesting strong separability. Overall, the matrix highlights strong classification ability.

4.4 Comparison with EfficientNet-B0

For EfficientNet-B0, the top layer of the pre-trained model was excluded to integrate custom layers tailored for the Bengali digit classification task. A dropout layer with a rate of 0.3 was introduced, followed by a dense layer utilizing ReLU activation and L2 regularization to prevent overfitting. An additional dropout layer was incorporated at the end for further regularization. The model was trained using a validation generator.

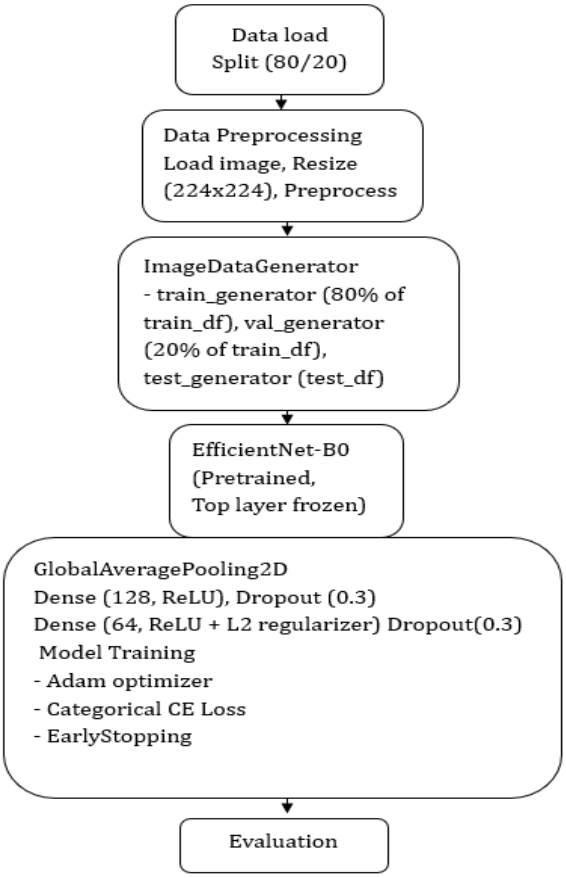

Figure 4: EfficientNet-B0 architecture

Table 8: EfficientNet-B0 classification on primary data

Class	Precision	Recall	F1-Score	Support
0	0.97	0.93	0.95	40
1	0.94	0.80	0.86	40
2	0.95	0.95	0.95	40
3	0.89	0.87	0.88	39
4	0.95	0.90	0.92	40
5	0.90	0.90	0.90	41
6	0.88	0.90	0.89	42
7	0.85	0.98	0.92	46
8	0.93	0.95	0.94	43
9	0.85	0.87	0.86	47
Accuracy			0.91	418
Macro Avg	0.91	0.91	0.91	418
Weighted Avg	0.91	0.91	0.91	418

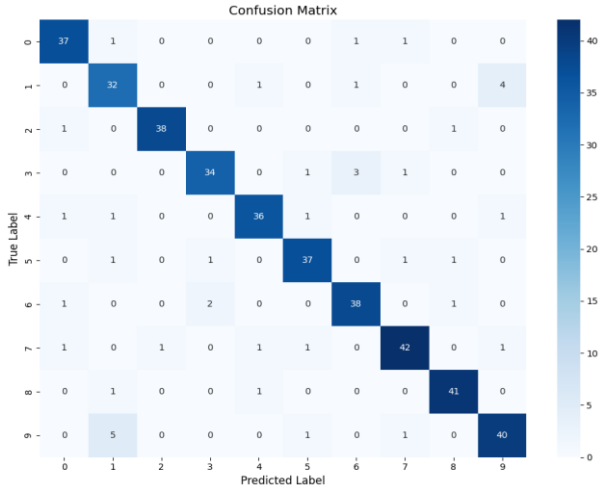

Figure 5: EfficientNet-B0 confusion matrix on primary data

The model recorded an overall accuracy of 91% on the test set, with macro-averaged precision, recall, and F1-score all at 0.91, demonstrating consistent performance across different classes. Most digits were classified with high precision and recall, reaching F1-scores exceeding 0.90 for digits 0, 2, 4, 5, 7, and 8, while slightly lower scores (0.86–0.89) were noted for digits 1, 3, 6, and 9, likely due to similarities within the classes or confusion with digits that share visual characteristics. The confusion matrix (figure 5) indicates that the predictions were predominantly aligned along the diagonal, underscoring strong overall performance, with only a few minor misclassifications primarily occurring among visually similar digits. Specifically, digit 1 was frequently misclassified as 9 (in five cases), and digit 3 exhibited some overlap with 6, while digit 7 recorded the highest recall with 42 correct identifications.

Table 9: EfficientNet-B0 classification report on NumtaDb

Class	Precision	Recall	F1-Score	Support
0	0.9901	0.9903	0.9902	1448
1	0.9382	0.9104	0.9241	1439
2	0.9619	0.9519	0.9569	1445
3	0.9403	0.9216	0.9309	1447
4	0.9911	0.9910	0.9910	1450
5	0.9548	0.9538	0.9543	1448
6	0.9284	0.9126	0.9204	1442
7	0.9896	0.9896	0.9896	1436
8	0.9790	0.9787	0.9788	1431
9	0.9312	0.9113	0.9211	1423
accuracy			0.9660	14409
macro avg	0.9605	0.9511	0.9557	14409
weighted avg	0.9659	0.9660	0.9659	14409

The model achieved a high overall accuracy of 97**%** (rounded) on the test set. The macro-averaged F1-score was 0.956, indicating consistent performance across all digit classes. Digits 0, 4, 7, and 8 were classified with highest precision and recall, while slightly lower scores were observed for digits 1, 3, 6, and 9 (F1-scores around 0.92), suggesting occasional confusion with visually similar classes. The results demonstrate the model's robustness and strong generalization ability over a large test set.

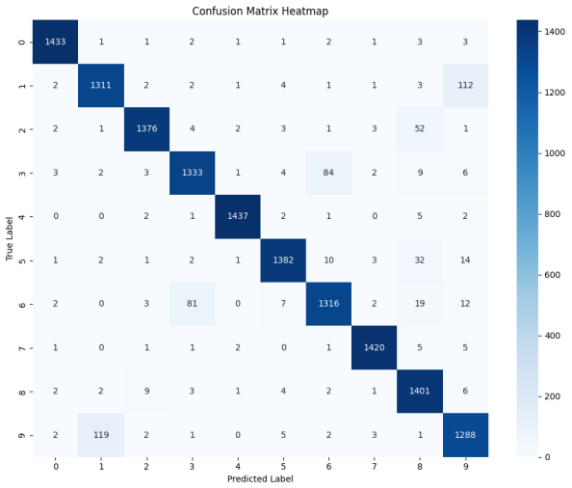

Figure 6: EfficientNet-B0 confusion matrix on NumtaDb

The confusion matrix indicates high overall classification accuracy, with most predictions concentrated along the diagonal. Digit classes 0, 4, 5, 7, and 8 show excellent performance with minimal misclassifications. However, notable confusion exists between certain pairs: Digit 1 is often misclassified as 9 (112 instances). Digit 3 has moderate confusion with 6 (84 instances). Digit 6 is occasionally predicted as 3 (81 instances). Digit 9 is misclassified as 1 in 119 cases.

A direct comparison reveals that both FusionNet and EfficientNet-B0 perform effectively in predicting Bengali numerals. While EfficientNet-B0 consistently achieves marginally higher accuracy (91% vs 87% on primary; 97% vs 96% on NumtaDb), it requires a longer execution time and more computational resources due to its deeper architecture. Conversely, FusionNet, despite a slight lag in peak accuracy, is a lightweight and faster model, requiring less execution time. For the FusionNet model, applying parallel processing reduced the overall processing time. On the custom dataset, the process took approximately 30 minutes without parallelization, while parallel processing brought it down to around 19 minutes, resulting in a speedup of 1.6×. Similarly, for the larger NumtaDb dataset, the processing time was reduced from about 2 hours without parallelization to just 1 hour with parallel processing, yielding a 2× improvement in efficiency. In contrast, the EfficientNet-B0 model, required approximately 35 minutes for the custom dataset and about 2.5 hours for the NumtaDb dataset. FusionNet likely

requires fewer floating-point operations (FLOPs) for each forward pass compared to deeper CNNs, as it utilizes pre-extracted handcrafted features along with a relatively shallow MLP. On the other hand, EfficientNet-B0 processes images through a deeper stack of convolutional and fully connected layers, resulting in a significantly higher computational demand and memory usage. The training and inference of EfficientNet-B0 also benefited from google colab's T4 GPU acceleration. Although FLOPs weren't directly quantified in this research, the decreased runtime noted during training and inference confirms the assumption that FusionNet is more computationally efficient in practice.

4.5 McNemar's test

The McNemar test in table 10 & 11 assesses whether there is a significant difference in predictions made by two models by examining the cases where they do not agree.

Table 10: McNemar's test result on custom dataset

Contingency Table	[[0, 22], [38, 0]]
McNemar's Test Statistic	3.75
P-value	0.054

Table 11: McNemar's Test Result on NumtaDb

Contingency Table	[[0, 179], [218,0]]
McNemar's Test Statistic	3.6372
P-value	: 0.0503

For both the custom dataset (p = 0.054) and the NumtaDb dataset (p = 0.0503), the p-values are above the 0.05 significance threshold. This indicates that the differences observed in their misclassifications are not statistically significant, leading us to fail to reject the null hypothesis that their performances are equal. Although the contingency tables reveal that each model has made mistakes on different instances, the extent of disagreement is insufficient to establish a genuine performance disparity. Hence, FusionNet and EfficientNet-B0 can be regarded as having statistically comparable performance on both datasets.

4.6 Visualization and further analysis

To further understand the discriminative power of FusionNet's feature representation, t-SNE (t-Distributed Stochastic Neighbor Embedding) visualizations were generated for both datasets.

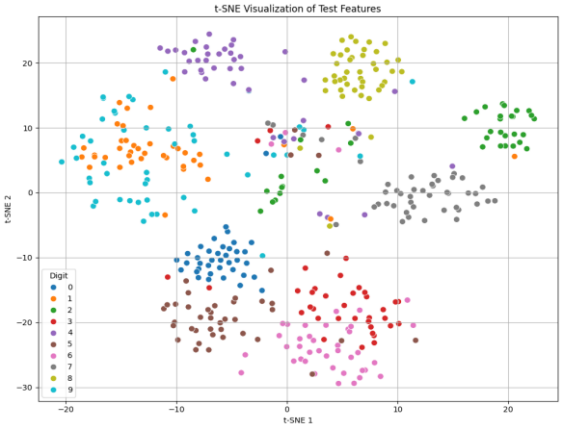

Figure 7: T-SNE visualization for FusionNet on primary dataset

Figure 7 illustrates the 2D t-SNE representation of FusionNet's acquired feature space on the main dataset. The visualization shows largely distinct and tightly packed clusters for each digit category (0–9), suggesting efficient learning of class-specific features. While minor overlaps are observed between similar classes such as 1 & 9 and 3 & 6, the overall distinctness indicates robust inter-class discriminability.

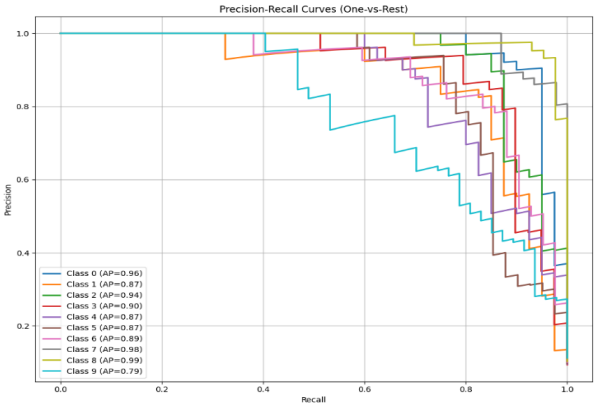

Figure 8: Precision-Recall for FusionNet on primary dataset

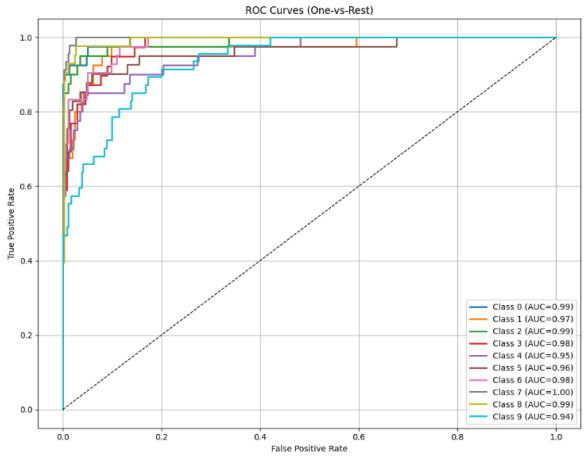

Figure 9: ROC Curves for FusionNet on primary dataset

In Figure 8 and 9, the Precision-Recall and ROC curves for FusionNet on the custom dataset are presented. The one-vs-rest PR curves reveal excellent precision and recall, as each class records an average precision exceeding 0.87. Additionally, the ROC curves highlight the model's robustness, with AUC values surpassing 0.95 for all classes, validating its impressive performance in multi-class classification.

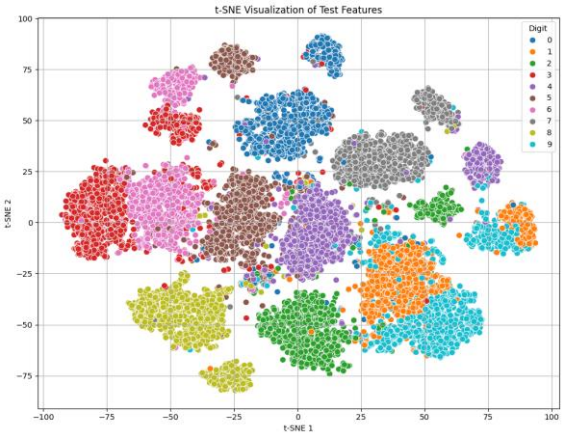

Figure 10: T-SNE visualization for FusionNet on NumtaDb dataset

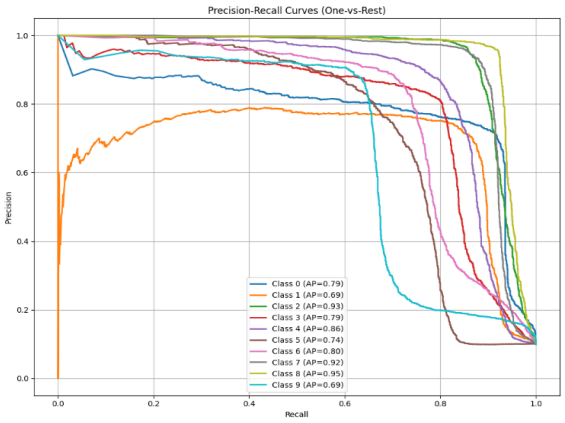

Figure 11: Precision-Recall curves for FusionNet on NumtaDb dataset

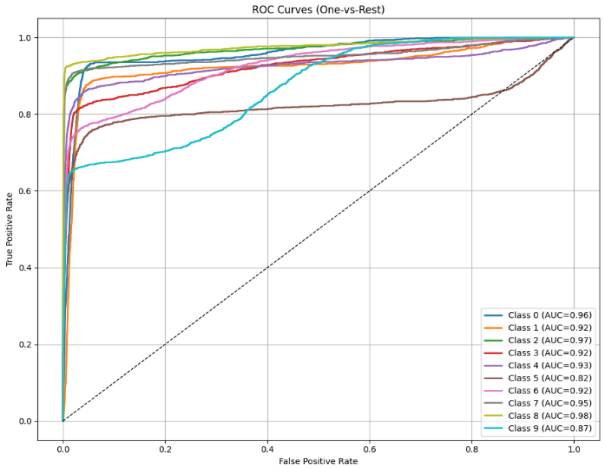

Figure 12: ROC Curves for FusionNet on NumtaDb dataset

Figures 10, 11 and 12 demonstrate the impressive feature learning and classification capabilities of FusionNet. The 2D t-SNE plot reveals distinctly separated and tightly grouped clusters for each digit class, showcasing a high level of inter-class distinction and intra-class uniformity. Both the Precision-Recall curves and ROC curves illustrate robust performance for all classes, with particularly high average precision (approximately 0.90) for Classes 2, 4, 7, and 8. Collectively, these findings validate that FusionNet's hybrid architecture successfully extracts distinguishing features, facilitating accurate recognition of Bengali handwritten digits.

5 Discussion

The FusionNet model, a proposal that has been advanced, has exhibited considerable proficiency in the identification of Bengali handwritten digits, attaining a commendable level of accuracy through the incorporation of handcrafted feature extraction techniques, namely Histogram of Oriented Gradients (HOG) and LBP, with a two-stage classification pipeline that integrates K-Nearest Neighbors (KNN) and a Multi-Layer Perceptron (MLP). Robust preprocessing techniques, including grayscale conversion, normalization, and Otsu's thresholding, contributed significantly to improving consistency and alignment, which in turn enhanced feature saliency. Compared to state-of-the-art models such as EfficientNet-B0 and SynergiProtoNet, FusionNet offers a compelling trade-off between classification performance and computational efficiency. Whilst EfficientNet-B0 demonstrated a marginally superior level of accuracy (97% compared with FusionNet's 96% on NumtaDb), it necessitated a substantially greater degree of computational resources. SynergiProtoNet, while effective in few-shot learning contexts, is heavily dependent on large-scale pretrained feature extractors and meta-learning strategies, which introduce architectural complexity and longer convergence times. In contrast, FusionNet attained competitive accuracy, underscoring its aptitude for environments with limited resources. The optimized parameter mechanism in the KNN stage facilitated adaptive initial predictions, which were subsequently refined by the MLP. This architecture not only mitigated overfitting but also ensured enhanced generalization across handwriting variations. The classification report and confusion matrix indicated balanced learning with minimal misclassification rates across classes. Nonetheless, a degree of confusion persisted between visibly similar digits, a problem with which CNNs are also confronted – suggesting the requirement for more discriminative features. To address this, the incorporation of additional feature extraction methods, such as shape descriptors and stroke-based representations, could be a viable solution. The enhancement of preprocessing through the incorporation of denoising, skeletonization, and adaptive thresholding has the potential to further reduce noise and enhance the

clarity of boundaries. Furthermore, the integration of ensemble methods, such as the combination of KNN with Recurrent Neural Networks (RNNs), has the potential to enhance temporal coherence in digit patterns and further robustness.

In summary, while FusionNet may exhibit slightly lower levels of raw accuracy in comparison to deep CNN-based models, it demonstrates notable strengths in terms of training speed, simplicity, and resource efficiency. These characteristics render it particularly advantageous for deployment in low-resource or embedded systems, where computational cost is a critical constraint.

6 Conclusion

This research introduced FusionNet, a compact hybrid architecture that integrates K-Nearest Neighbors (KNN) and Multi-Layer Perceptron (MLP) in a two-stage classification pipeline, utilizing complementary handcrafted features (HOG, LBP) to attain effective and precise recognition of Bengali handwritten digits. Results from experiments conducted on a custom dataset and the standard NumtaDb dataset show that FusionNet achieves competitive results: 87% and 96% accuracy, respectively. With substantially lower computational demands than state-of-the-art deep models such as EfficientNet-B0. While EfficientNet-B0 reached slightly superior accuracy, McNemar's test indicated no statistically significant difference between the performance of the models, emphasizing the robustness and generalization ability of FusionNet. By incorporating parallel processing to decrease overall runtime duration and utilizing meticulous preprocessing to improve feature quality, FusionNet demonstrates exceptional suitability for use in resource-limited settings. Future efforts might aim at incorporating more discriminative features, enhancing preprocessing techniques, and examining ensemble methods to further boost recognition precision and robustness against visually alike digits.

Aknowledgement

NumtaDb dataset is publicly available on kaggle: https://www.kaggle.com/datasets/BengaliAI/numta

Full code can be found in this link: https://github.com/A-H-Sumon/FusionNet

References

[1] Dalui, Abhraneel, Rahul Sarkar, Suvam Sharma, Akash Ghosh, Sheryl Brahnam, and Satya Ranjan Dash. "A Deep Convolutional neural network approach to recognize Bangla handwritten digits." In *2024 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC)*, pp. 1-5. IEEE, 2024. DOI: 10.1109/ASSIC60049.2024.10507895

[2] Azgar, Ali, Md Imran Nazir, Afsana Akter, Md Saddam Hossain, Md Anwar Hussen Wadud, and Md Reazul Islam. "MNIST handwritten digit recognition using a deep learning-based modified dual input convolutional neural network (DICNN) model." In *International Congress on Information and Communication Technology*, pp. 563-573. Singapore: Springer Nature Singapore, 2024. https://doi.org/10.1007/978-981-97-3562-4_44

[3] Ali, Ashikin, Norhalina Senan, and Norhanifah Murli. "Convolutional neural network using regularized conditional entropy loss (CNNRCoE) for MNIST handwritten digits classification." In *International Conference on Soft Computing and Data Mining*, pp. 337-348. Cham: Springer Nature Switzerland, 2024. https://doi.org/10.1007/978-3-031-66965-1_33

[4] Parihar, Giriraj, Ratnavel Rajalakshmi, and J. Bhuvana. "Multi-Lingual Handwritten Character Recognition Using Deep Learning." *Computational Analysis and Deep Learning for Medical Care: Principles, Methods, and Applications* (2021): 155-180. DOI:10.1002/9781119785750.ch7

[5] Khudeyer, Raidah Salim, and Noor Mohammed Almoosawi. "Combination of machine learning algorithms and Resnet50 for Arabic Handwritten Classification." *Informatica* 46, no. 9 (2023). https://doi.org/10.31449/inf.v46i9.4375

[6] Chatterjee, Swagato, Rwik Kumar Dutta, Debayan Ganguly, Kingshuk Chatterjee, and Sudipta Roy. "Bengali handwritten character classification using transfer learning on deep convolutional network." In *International Conference on Intelligent Human Computer Interaction*, pp. 138-148. Cham: Springer International Publishing, 2019. https://doi.org/10.1007/978-3-030-44689-5_13

[7] Akhand, M. A. H., Mahtab Ahmed, and M. M. Rahman. "Convolutional Neural Network based Handwritten Bengali and Bengali-English Mixed Numeral Recognition." *International Journal of Image, Graphics & Signal Processing* 8, no. 9 (2016). DOI: 10.5815/ijigsp.2016.09.06

[8] Sufian, Abu, Anirudha Ghosh, Avijit Naskar, Farhana Sultana, Jaya Sil, and MM Hafizur Rahman. "Bdnet: bengali handwritten numeral digit recognition based on densely connected convolutional neural networks." *Journal of King Saud University-Computer and Information Sciences* 34, no. 6 (2022): 2610-2620. https://doi.org/10.1016/j.jksuci.2020.03.002

[9] Maity, Suprabhat, Anirban Dey, Ankan Chowdhury, and Abhijit Banerjee. "Handwritten Bengali character recognition using deep convolution neural network." In *International Conference on Machine Learning, Image Processing, Network Security and Data Sciences*, pp. 84-92. Singapore: Springer Singapore, 2020. https://doi.org/10.1007/978-981-15-6318-8_8

[10] Amin, Ruhul, Md Shamim Reza, Yuichi Okuyama, Yoichi Tomioka, and Jungpil Shin. "A fine-tuned hybrid stacked cnn to improve bengali handwritten

digit recognition." *Electronics* 12, no. 15 (2023): 3337. https://doi.org/10.3390/electronics12153337

[11] Azad, Md Ali, Hijam Sushil Singha, and Md Mahadi Hasan Nahid. "Bangla handwritten character recognition using deep convolutional autoencoder neural network." In *2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT)*, pp. 295-300. IEEE, 2020. DOI: 10.1109/ICAICT51780.2020.9333472

[12] Mondal, Sudarshan, and Nagib Mahfuz. "Convolutional neural networks based bengali handwritten character recognition." In *International Conference on Cyber Security and Computer Science*, pp. 718-729. Cham: Springer International Publishing, 2020. https://doi.org/10.1007/978-3-030-52856-0_57

[13] Purkaystha, Bishwajit, Tapos Datta, and Md Saiful Islam. "Bengali handwritten character recognition using deep convolutional neural network." In *2017 20th International conference of computer and information technology (ICCIT)*, pp. 1-5. IEEE, 2017. DOI: 10.1109/ICCITECHN.2017.8281853

[14] Bappi, Jabed Omor, and Mohammad Abu Tareq Rony. "CBD2023: A Hypercomplex Bangla Handwriting Character Recognition Data for Hierarchical Class Expansion." *Data in Brief* 52 (2024): 109909. https://doi.org/10.1016/j.dib.2023.109909

[15] Shawon, Ashadullah, Md Jamil-Ur Rahman, Firoz Mahmud, and MM Arefin Zaman. "Bangla handwritten digit recognition using deep CNN for large and unbiased dataset." In *2018 international conference on Bangla speech and language processing (ICBSLP)*, pp. 1-6. IEEE, 2018. DOI: 10.1109/ICBSLP.2018.8554900

[16] Alam, Samiul, Tahsin Reasat, Rashed Mohammad Doha, and Ahmed Imtiaz Humayun. "Numtadb-assembled bengali handwritten digits." *arXiv preprint arXiv:1806.02452* (2018). https://doi.org/10.48550/arXiv.1806.02452

[17] Deng, Li. "The mnist database of handwritten digit images for machine learning research [best of the web]." *IEEE signal processing magazine* 29, no. 6 (2012): 141-142. DOI: 10.1109/MSP.2012.2211477

[18] Ahamed, Mehedi, Radib Bin Kabir, Tawsif Tashwar Dipto, Mueeze Al Mushabbir, Sabbir Ahmed, and Md Hasanul Kabir. "Performance Analysis of Few-Shot Learning Approaches for Bangla Handwritten Character and Digit Recognition." In *2024 6th International Conference on Sustainable Technologies for Industry 5.0 (STI)*, pp. 1-6. IEEE, 2024. DOI: 10.1109/STI64222.2024.10951048

[19] Tan, Mingxing, and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks." In *International conference on machine learning*, pp. 6105-6114. PMLR, 2019.

[20] Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database." In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248-255. Ieee, 2009. DOI: 10.1109/CVPR.2009.5206848

[21] Chaudhari, Shailesh A., and Ravi M. Gulati. "An OCR for separation and identification of mixed English—Gujarati digits using kNN classifier." In *2013 International Conference on Intelligent Systems and Signal Processing (ISSP)*, pp. 190-193. IEEE, 2013. DOI: 10.1109/ISSP.2013.6526900

[22] Matei, Oliviu, Petrica C. Pop, and H. Vălean. "Optical character recognition in real environments using neural networks and k-nearest neighbor." *Applied intelligence* 39, no. 4 (2013): 739-748. https://doi.org/10.1007/s10489-013-0456-2

[23] Zhang, Xu, Feihong Li, Chenlong Li, and Xufeng Yu. "Chinese Medical Named Entity Recognition Using Pre-Trained Language Models and an Efficient Global Pointer Mechanism." *Informatica* 49, no. 19 (2025). https://doi.org/10.31449/inf.v49i19.8043