2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 979-8-3503-7024-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICCCNT61001.2024.10726018

Multilingual Handwritten Digit Recognition Using Multiplexer-Based Deep Learning Models

1st Amaan Mansuri
Dept. of Comp. Sci. & Engg.
Nirma University
Ahmedabad, India
20bce011@nirmauni.ac.in

2nd Atir Sakhrelia
Dept. of Comp. Sci. & Engg.
Nirma University
Ahmedabad, India
20bce022@nirmauni.ac.in

3rd Anaya Patel
Dept. of Comp. Sci. & Engg.
Nirma University
Ahmedabad, India
20bce013@nirmauni.ac.in

4th Priyank Thakkar
Dept. of Comp. Sci. & Engg.
Nirma University
Ahmedabad, India
priyank.thakkar@nirmauni.ac.in

5th Ankit Sharma
Dept. of Electronics & Instrumentation
Nirma University
Ahmedabad, India
ankit.sharma@nirmauni.ac.in

6th Krishn Limbachiya
Dept. of Electronics & Instrumentation
Nirma University
Ahmedabad, India
20ftphde43@nirmauni.ac.in

***Abstract*—Recognizing handwritten digits poses a significant challenge in machine learning and image processing due to the inherent variations in individual writing styles, sizes, curves, strokes, and interpretations. Using deep learning models, the research presented here describes an innovative method for classifying handwritten digits in various languages, including Urdu, Gujarati, Hindi, and Bengali. The approach proposed utilizes a model that employs a multiplexing concept to build a more efficient and scalable model as compared to a naive 50-class classifier model capable of distinguishing between the digits. The use of a multiplexer-based approach streamlines the classification process and achieves 98.88% accuracy, offering a promising solution for recognizing handwritten digits in diverse linguistic contexts.**

***Index Terms*—Deep Learning, Convolution Neural Networks, Transfer Learning, Multilingual Handwritten Digit Recognition, Multiplexer**

I. Introduction

Handwritten digit recognition is the process of recognizing and interpreting handwritten numerical digits. It is a challenging problem in the field of machine learning and computer vision because handwritten digits can vary significantly in terms of writing style, size, curves, strokes, thickness, interpretation, pivot, and twisting. Unlike printed characters, which follow predefined patterns and structures, handwritten digits are more diverse and subjective.

The main objective of Handwritten digit recognition is to create algorithms and systems that can accurately transform handwritten digits into machine-readable representations. Various applications can benefit from automating the recognition process, including postal code recognition, cheque processing, form filling, signature verification, and digitization of historical handwritten documents. These applications rely on the ability to read and comprehend handwritten numerical data. Accurate digit identification is crucial. The ability to correctly recognize handwritten numerals is essential for enabling secure and reliable procedures in various fields.

Researchers use a variety of strategies to address the issues of handwritten digit identification, including the development and training of models that can learn patterns and characteristics from a huge dataset of handwritten digits. Common approaches include deep learning methods like convolutional neural networks (CNNs), which have demonstrated excellent performance in digit recognition challenges. In recent years, there has been a rise of interest in developing deep learning models capable of identifying handwritten digits in a variety of languages, including Bengali, English, Kannada, Devanagari, Farsi, Persian, Arabic, and Mandarin.

Although there has been a significant amount of study on recognizing handwritten numbers in a single script or language, there have been very few studies concentrating on handwritten numeral recognition across several scripts. The model proposed in this paper demonstrates a successful classification of digits in languages such as Urdu, Gujarati, Hindi, Bengali, and English, often used in the Indian subcontinent. To address this challenge, we have proposed a unique "Multiplexer Model" as a feasible solution.

Previous work done on multilingual digit classification using a Convolution Model by Yihan Wang [12] provided a comparative evaluation of two different CNN models trained and tested on 15 different languages. The majority of other research done to address the challenge of digit recognition employs the concept of transfer learning and ensemble the features, extracted using many state-of-the-art Deep learning models [1]. Other relevant studies generally used the concept of merging the digits which do have similar shapes into a single class and hence consequently decreasing the total number of classes [13].

II. Related Work

Weiwei Jiang and Le Zhang [6] proposed two models Edge-Siamnet and Edge-TripleNet. The canny edge extraction approach is largely used in both models to extract edges from images and using the Siamese or Triple network on the original image as well as on the edge image to recognize the digit present in the input image. Edge SiamNEt and Edge-TripleNet obtained 99.36% and 99.26% accuracy, respectively, on the MNIST dataset. Ahmad Al-Hmouz et al. [3] proposed a digit recognition system using intends to identify the fuzzy set of features from the extracted features on digits of English, Arabic, Persian, and Devanagari languages.

Farhan Shahid et al. [15] focused solely on using machine learning (ML) based algorithms to achieve and compare notable accuracies. The SVM model had the highest accuracy (99.7%). Abu Sufian et al. [14] provided a new data set as a testing dataset of 1000 Bengali handwritten numbersx. The proposed model achieved a high test accuracy of 99.78% on the ISI Bengali handwritten numeral dataset. This was owing to the use of BDNet, a densely linked convolutional neural network that is a task-oriented model specifically created for recognizing Bengali handwritten numerals. In a study conducted by Rasoul Ameri et al. [11], the influence of data pre-processing and fine-tuning the network architecture on performance was investigated. As a result, an accuracy of 99.1% was achieved in the classification task for Chinese handwritten numbers.

Alkhateeb et al [9] The proposed Convolutional model had 3 convolution layers, 2 max-pooling layers, and a fully connected layer, resulting in a 94.3% accuracy on the ADbase dataset. Weiwei Jiang [5] an MNIST-MIX dataset is created by combining 13 separate datasets in ten languages, including Arabic, Bengali, Devanagari, English, Farsi, Kannada, Swedish, Telugu, Tibetan, and Urdu. The authors applied a LeNet model to this dataset and achieved an accuracy of 90.34%. Yihan Wang [12] evaluated the performance of two CNN models, one a LeNet model and the other a more advanced CNN model, was evaluated on the biggest multilingual handwritten numeral dataset, which included languages such as Swedish, Bengali, Devanagari, Arabic, Telugu, English, Farsi, Kannada, and others. The paper concluded that the LeNet model tends to outperform the more complex CNN model in terms of its effectiveness. All the images in the dataset were preprocessed to be in a grayscale format and converted to a size of 28 x 28 through bilinear interpolation. The LeNet model had an accuracy of 95.54% while the CNN model had an accuracy of 94.87%.

Alkhawaldeh et al. [4] suggested an ensemble deep learning architecture for Arabic digit classification. They proposed a combination of two pre-trained models namely ResNet50 and MobileNetV2. The accuracy of 99.83% was achieved on the dataset of Arabic digits. In [10], Alkhawaldeh et al. proposed combining two pre-trained models, GoogleNet and VGG-16, followed by an LSTM and a fully linked dense layer. Pretrained models extracted features, while the LSTM network learned more critical features from them. They achieved an accuracy of 98.92%. Basri, Rabeya, et al. [8] gave an overall analysis of using multiple state-of-the-art Convolutional models, including AlexNet, MobileNet, GoogLeNet, and CapsuleNet models, to execute the task of digit recognition in Bengali. These four deep CNN models were evaluated on an enormous, impartial, and significantly expanded standard dataset called NumtaDB, and the article determined that AlexNet performed the best in terms of precision and computation time.

Vidhale, Bhushan, et al. [2] introduced the largest multilingual handwritten dataset for digits. It uses a combined dataset of 15 languages. The accuracy achieved for the LeNet model is 93.73% and the CNN model is 89.95%. The paper presented by Hazrat Ali et al. [7], they provide a pioneer dataset of handwritten Urdu numerals and letters. They created a self-learning recognition model with an intricate autoencoder and a convolutional network. They implemented a 3-layer and 2-layer autoencoder and convolutional network and had an accuracy of 97.3% and 96.8% respectively.

III. Proposed Architecture

A. Dataset Description

In our proposed approach, our main focus is on the multilingual classification task of Indian languages. To achieve this, the authors utilized handwritten digit images from five languages: Hindi, Urdu, Gujarati, Bengali, and English.

Each dataset for the individual language digit recognition challenge was divided into three parts, 70:20:10 ratio (Train:Validation:Test). Additionally, for training the Language recognition model, the creation of an amalgamated dataset was done by combining all the images from all languages.

TABLE I
Datasets and their Distribution

Language	Training set	Validation set	Testing set	Total
Bengali	10,929	3,120	1,571	15,620
English	29,394	8,396	4,210	42,000
Gujarati	12,603	3,599	1,807	18,009
Hindi	15,703	4,485	2,251	22,439
Urdu	6,606	1,987	1,414	10,007

Initially, all the images were grayscale images and had sizes 28x28.To train and evaluate these deep learning models, it is crucial to have high-quality datasets that contain a diverse set of handwritten digits. Consequently, preprocessing was done to ensure uniformity across all datasets. Since each model was trained on images of size 32x32, any images smaller than this size were upsampled to 32x32 using bilinear interpolation techniques. Normalization was performed on all images by dividing each pixel value by 255.

From a total of 108,075 pictures, the authors used 75,235 for training, 21,587 for validation, and 11,253 for testing. Table 1 summarizes the languages used, the dataset origin, and the total number of images utilized for training, validation, and testing.

B. Proposed Approach

The authors developed a comprehensive technique to address the classification of handwritten digits based on their

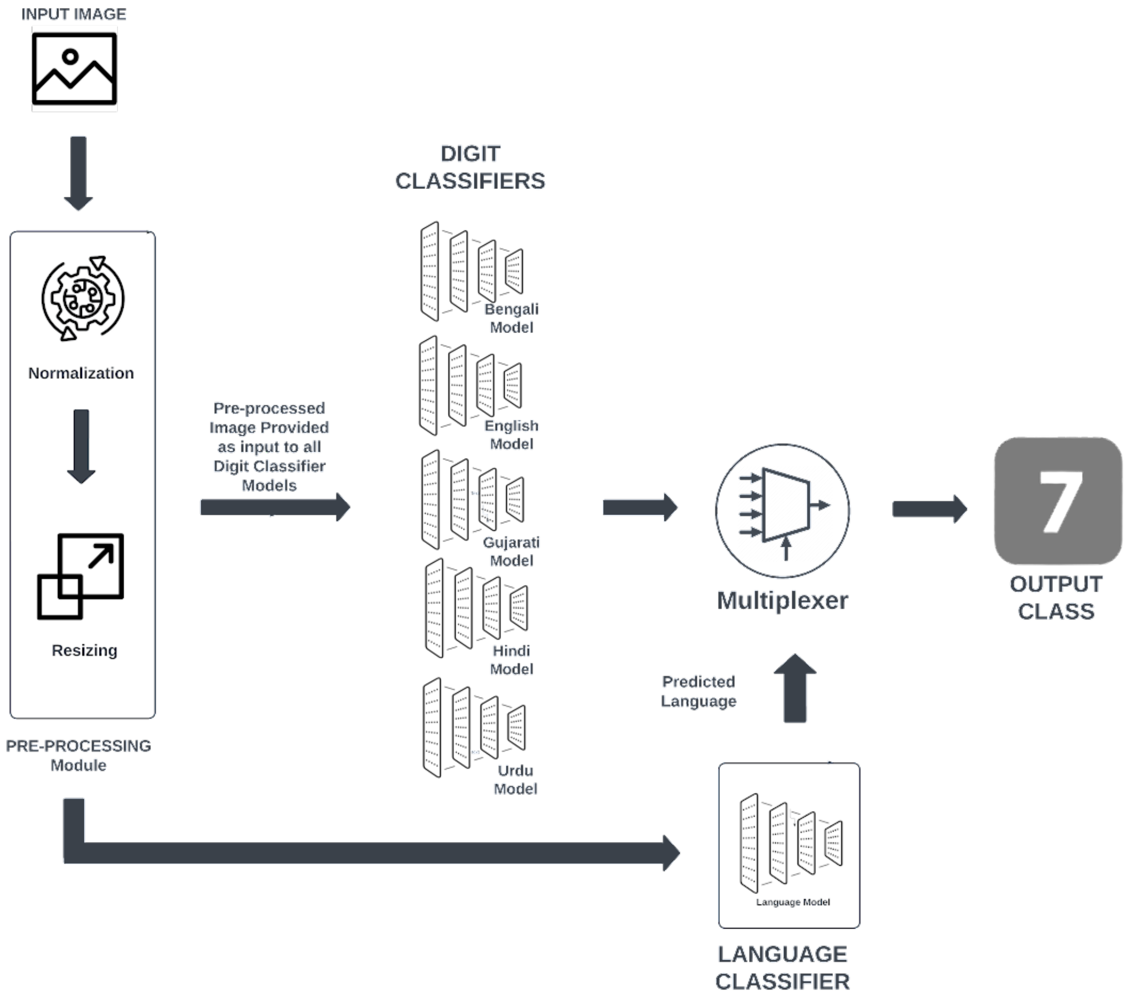

Fig. 1. **Proposed Architecture**

language of origin. The proposed approach involves two stages:

Training five language-specific models tailored for Hindi, Urdu, Gujarati, Bengali, and English, using datasets specific to each language. Training a language classifier model on a combined dataset containing handwritten digits from all five languages. This model identifies the language associated with a digit.

The system functions akin to a Multiplexer, where a language model acts as the selection line to choose the appropriate Digit Classifier for a given input digit, thereby enhancing prediction accuracy. This configuration is termed the ”Multiplexer Model”.The visual representation of our proposed approach is shown in Figure 1.

The input image is simultaneously processed by all five digit classifier models and the language classification model. The language classifier model generates predictions for each of the five languages, indicating the likelihood of the input belonging to each language. The overall architecture selects the language with the highest probability, then chooses the corresponding digit classifier model with the highest probability output for the final prediction. This approach optimizes prediction accuracy by leveraging separate models and adjusting the final output based on language categorization.

Extensive experiments and model fine-tuning are conducted to ensure adaptability and effectiveness across diverse Indian languages.This enables the method to handle the complexities and subtleties of recognizing handwritten numerals in varied language settings. Algorithm 1 outlines the pipeline of the multiplexer technique.

C. Architecture

Deep learning architectures have undeniably asserted their dominance as powerful tools in the realm of classification, showcasing remarkable efficacy and sophistication in discerning and categorizing diverse patterns and datasets. The authors utilized the same CNN model for both the language recognition and digit recognition phases because of the comparable structure of the input images, with minor adjustments in specific layers. The architecture consists of three convolutional

Algorithm 1 Multiplexer Model for Digit Recognition

Standardize input size of M_i to $32 \times 32 \times 1$
Upsample non-matching input sizes using interpolation
Normalize Image By Diving each pixel with 255.0
procedure TRAINMODELS
    for $i \leftarrow Languages$ **do**
        Train digit classifier model M_i
    end for
    Train language classifier model L
end procedure
procedure MULTIPLEXERPREDICTION(input_image)
    for $i \leftarrow Languages$ **do**
        $digit\_prediction_i \leftarrow M_i$(input_image) ▷ Digit classifier predictions
    end for
    $language\_predictions \leftarrow L$(input_image) ▷ Language classifier predictions
    $selected\_language \leftarrow \arg\max(language\_predictions)$ ▷ Identify language
    $final\_prediction \leftarrow M_{\text{selected\_language}}$(input_image) ▷ Final digit prediction
    return $final\_prediction$
end procedure

blocks, a flattening layer, fully linked layers, and an output layer.

The neural network comprises of multiple convolutional blocks followed by fully connected layers. For every convolutional block, Conv2D layers are followed by batch normalization and max-pooling operations. The number of kernels varies across blocks: 6 kernels in the first block, 16 in the next one and 120 in the last one. ReLU is used as the activation function, and the size of the kernels is 5 x 5.

After convolutional blocks the feature map is flattened and then goes through fully connected layers. The first layer is a fully connected layer with 120 neurons and ReLU activation function, followed by batch normalization. Thereafter there are fully connected layers with 84 neurons, also normalized. The output layer uses Softmax activation with 5 neurons for the Language classifier model and 10 neurons for the Digit classifier model, depending on the given classification task.

1) Language Classifier: The fundamental step in our research is to create a language recognition model that can accurately identify digits to their respective languages. A previous study has demonstrated that applying a digit recognition model directly to multi-language texts generates poor results and reduced classification accuracy. To overcome this issue, we propose using a language recognition model to categorize photos and then leveraging the output, resulting in the final anticipated output for the given input. The aforementioned technique allows us to determine the proper digit recognition based on the recognized language. Furthermore, adopting a language recognition module fosters the system's scalability and resilience, enabling it to handle new scripts with different languages without having to retrain the whole digit recognition model.

TABLE II
FINAL EVALUATION MATRIX

Model	Accuracy	Precision	Recall	F1 Score
Bengali Classifier	97.89%	97.92%	97.89%	97.90%
English Classifier	99.16%	99.17%	99.16%	99.16%
Gujarati Classifier	99.72%	99.72%	99.72%	99.72%
Hindi Classifier	99.64%	99.64%	99.65%	99.64%
Urdu Classifier	97.80%	97.82%	97.80%	97.80%
50 Class Model	98.25%	98.26%	98.25%	98.25%
Language Classifier	99.63%	99.63%	99.63%	99.63%
Multiplexer Model	98.88%	98.88%	98.88%	98.88%

2) Digit Classifier: Following an accurate determination of the language of the input image, the next critical step is digit recognition. For digit recognition in all languages, the structure of our proposed model remains consistent. However, in accordance with the outcome of the language recognition module, we train an independent instance of the model for each of the languages. The incorporation of the language recognition module in our approach encompasses various advantages. A significant benefit is the automated selection of the appropriate digit recognition mode.

IV. RESULTS AND DISCUSSION

This section explains the results and performance of the suggested architecture for multilingual digit classification. A summary explanation is provided below.

A. Evaluation Metrics

We ran a series of assessments utilizing all five handwritten languages to evaluate and further improve the effectiveness of our proposed methodology. Several metrics, such as accuracy, precision, and F1-score, were utilized to evaluate the performance of both individual digit classifier models and the Multiplexer Model.

Figures 2 and 3 show visual representations of the model accuracy and loss over epochs, as well as the confusion matrix for all the five individual digit classifier models and the language classifier model. Figure 4 shows the confusion matrix of the proposed Multiplexer Model.

Table II shows the Accuracy, Precision, Recall, and F1-score of all the models used by us in our proposed method.

Based on the evaluation, our suggested approach beat both transfer learning models and the naive 50-class classification model in terms of accuracy and performance. As a result, our proposed approach proved to be more effective in addressing the challenge of multilingual digit recognition.

B. Discussion

Other Approaches: To evaluate and compare our proposed model against standard deep learning approaches, we implemented a transfer learning model and a naive 50-class classifier model. The goal was to assess the accuracy and performance of these models in comparison to our proposed approach.

The project implemented transfer learning using ResNet50 and MobileNetV2 models for Hindi, Gujarati, and Bengali

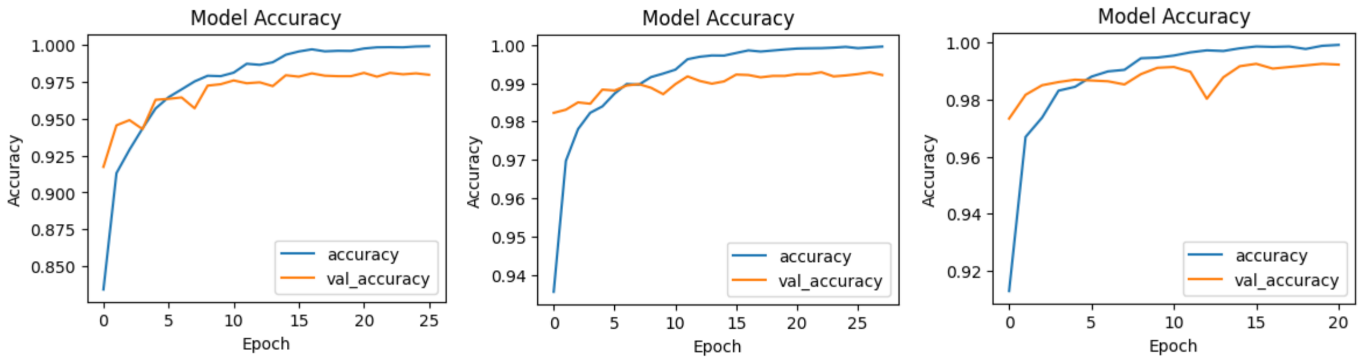

Fig. 2. **Model Evaluation for Bengali, English, and Gujarati**

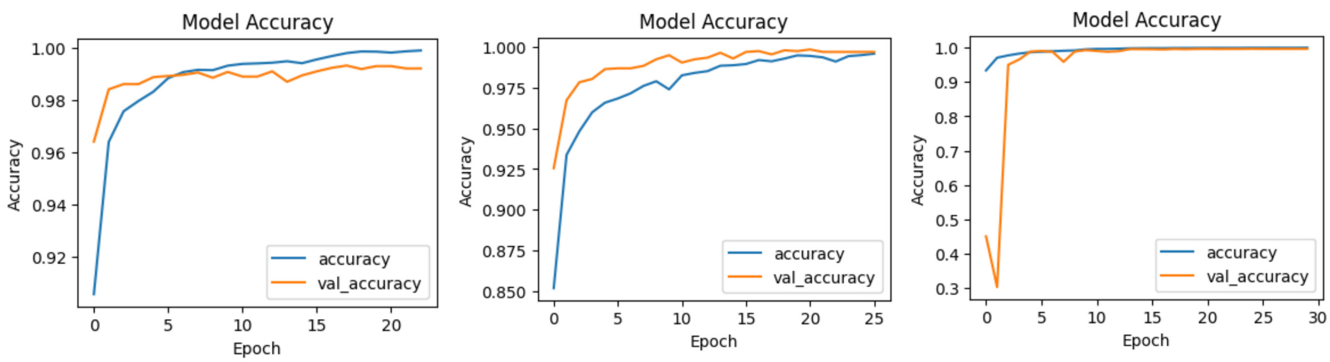

Fig. 3. **Model Evaluation for Hindi, Urdu, and Langauge**

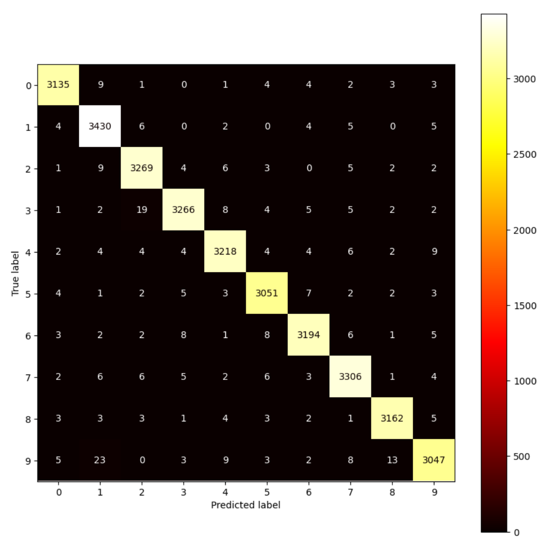

Fig. 4. **Confusion Matrix of the ProposedMultiplexer Architecture**

digit recognition. However, the achieved accuracies (95.65%, 96.68%, and 91.98% for ResNet50, and 54.73%, 74.05%, and 61.36% for MobileNetV2) were lower than those obtained with the our model, leading to the decision not to use transfer learning for the problem.

Additionally, a naive 50-class classification model was explored, sharing the architecture with digit and language classifiers but with 50 output neurons (10 for each language). This model demonstrated an accuracy of about 98.25%, outperforming the transfer learning models and highlighting its effectiveness in the context of the project.

After evaluating the proposed architecture, our system outperformed the existing models along with the naive 50-class classification model. The currently existing models tends to achieve high accuracy on individual languages but fail to do so in multi-language classification. The accuracy (98.88%) achieved by our multiplexer model has not been yet achieved by any model.

V. Conclusion

The proposed method effectively demonstrated the effectiveness of deep learning, specifically Convolutional Neural Networks (CNNs), in detecting handwritten numbers across several languages. Achieving an accuracy of 98.88%, the project introduced the innovative Multiplexer method to address language dependency. By inputting images into models corresponding to different languages, the Multiplexer method leveraged language-specific patterns, overcoming limitations of single-language models and enhancing recognition accuracy.

The proposed Multiplexer Model and other methodologies and sophisticated techniques like data augmentation and transfer learning can help in increasing model accuracy. The project's outcomes extend to various applications such as optical character recognition, signature verification, and digitizing historical manuscripts.

As future work, scaling up the project to include additional languages is suggested, making the recognition problem more challenging and applicable to real-world scenarios. Incorporating more languages would expand the number of classes, forcing the model to learn a broader range of features and patterns, ultimately increasing generalization to new samples.

REFERENCES

[1] Fateh, Amirreza, Mansoor Fateh, and Vahid Abolghasemi. "Multilingual handwritten numeral recognition using a robust deep network joint with transfer learning." Information Sciences 581 (2021): 479-494.

[2] Vidhale, Bhushan, et al. "Multilingual Text and Handwritten Digit Recognition and Conversion of Regional languages into Universal Language Using Neural Networks." 2021 6th International Conference for Convergence in Technology (I2CT). IEEE, 2021.

[3] Al-Hmouz, Ahmad, et al. "Enhanced numeral recognition for handwritten multi-language numerals using fuzzy set-based decision mechanism." International Journal of Machine Learning and Computing 10.1 (2020): 99-107.

[4] Alkhawaldeh, Rami S., et al. "Ensemble deep transfer learning model for Arabic (Indian) handwritten digit recognition." Neural Computing and Applications 34.1 (2022): 705-719.

[5] Jiang, Weiwei. "MNIST-MIX: a multi-language handwritten digit recognition dataset." IOP SciNotes 1.2 (2020): 025002.

[6] Jiang, Weiwei, and Le Zhang. "Edge-siamnet and edge-triplenet: New deep learning models for handwritten numeral recognition." IEICE Transactions on Information and Systems 103.3 (2020): 720-723.

[7] Ali, Hazrat, et al. "Pioneer dataset and automatic recognition of Urdu handwritten characters using a deep autoencoder and convolutional neural network." SN Applied Sciences 2 (2020): 1-12.

[8] Basri, Rabeya, et al. "Bengali handwritten digit recognition using deep convolutional neural network." Proceedings of the international conference on computing advancements. 2020.

[9] Alkhateeb, Jawad H. "Handwritten arabic digit recognition using convolutional neural network." International Journal of Communication Networks and Information Security 12.3 (2020): 411-416.

[10] Alkhawaldeh, Rami S. "Arabic (Indian) digit handwritten recognition using recurrent transfer deep architecture." Soft Computing 25.4 (2021): 3131-3141.

[11] Ameri, Rasoul, et al. "Classification of handwritten chinese numbers with convolutional neural networks." 2021 5th International Conference on Pattern Recognition and Image Analysis (IPRIA). IEEE, 2021.

[12] Wang, Yihan. "Multi-language Handwritten Numeral Recognition with Convolutional Neural Network." 2021 2nd International Conference on Computing and Data Science (CDS). IEEE, 2021.

[13] Gupta, Deepika, and Soumen Bag. "CNN-based multilingual handwritten numeral recognition: A fusion-free approach." Expert Systems with Applications 165 (2021): 113784.

[14] Sufian, Abu, et al. "Bdnet: Bengali handwritten numeral digit recognition based on densely connected convolutional neural networks." Journal of King Saud University-Computer and Information Sciences 34.6 (2022): 2610-2620.

[15] Farhan Shahid, Muhammad. "Influence of various ML-Based Binary Classifiers on the Performance on handwritten digit recognition." (2023).