Research Article

A Novel Technique for Handwritten Digit Recognition Using Deep Learning

Syed Sohail Ahmed [ID],[1] **Zahid Mehmood** [ID],[2,3] **Imran Ahmad Awan** [ID],[1] **and Rehan Mehmood Yousaf** [ID][4]

[1]*Department of Computer Engineering, College of Computer, Qassim University, Buraydah, Saudi Arabia*
[2]*Department of Computer Engineering, University of Engineering and Technology, Taxila 47050, Pakistan*
[3]*The FAMLIR Group, The University of Lahore, Lahore 54000, Pakistan*
[4]*University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University, Rawalpindi 44000, Pakistan*

Correspondence should be addressed to Zahid Mehmood; zahid.mehmood@uettaxila.edu.pk

Handwritten digit recognition (HDR) shows a significant application in the area of information processing. However, correct recognition of such characters from images is a complicated task due to immense variations in the writing style of people. Moreover, the occurrence of several image artifacts like the existence of intensity variations, blurring, and noise complicates this process. In the proposed method, we have tried to overcome the aforementioned limitations by introducing a deep learning- (DL-) based technique, namely, EfficientDet-D4, for numeral categorization. Initially, the input images are annotated to exactly show the region of interest (ROI). In the next phase, these images are used to train the EfficientNet-B4-based EfficientDet-D4 model to detect and categorize the numerals into their respective classes from zero to nine. We have tested the proposed model over the MNIST dataset to demonstrate its efficacy and attained an average accuracy value of 99.83%. Furthermore, we have accomplished the cross-dataset evaluation on the USPS database and achieved an accuracy value of 99.10%. Both the visual and reported experimental results show that our method can accurately classify the HDR from images even with the varying writing style and under the presence of various sample artifacts like noise, blurring, chrominance, position, and size variations of numerals. Moreover, the introduced approach is capable of generalizing well to unseen cases which confirms that the EfficientDet-D4 model is an effective solution to numeral recognition.

1. Introduction

Character recognition of various languages is highly explored by researchers [1–3]; however, the most prominent area is HDR. HDR plays a fundamental part in the area of information manipulation as a huge amount of data is available in the form of printed text or pictures [4]. Furthermore, analyzing digital information is more cost-effective than processing information manually from the printed paper. The objective of HDR approaches is to recognize and translate the handwritten digits into machine-understandable presentations. Nowadays, HDR grasps the extensive focus of the research community because of its multiple applications. These frameworks are capable of recognizing what is

written on printed pages and allow scientists to explore significant information saved on historic documents and files, which appears unrecognizable to human eyes [5]. Furthermore, HDR techniques are important for the digital revolution of any business and institute. Computerized HDR frameworks can assist in many areas, for example, automated detection of number plates of vehicles [6] and recognition of the digits written on medical receipts which can aid the chemists, patients, and staff. Moreover, psychologists can use HDR systems to analyze the patient's personality [7]. However, the above-mentioned areas have large databases; that is why automated HDR systems need to be efficient and effective with a small execution time and consistent results. In recent years, several HDR systems have been

presented to assist in a variety of applications. Such systems have the demand of exhibiting improved numeral recognition and classification accuracy in a consistent manner [6–8]. Automated recognition of historical handwritten scripts is a challenging task because of the varying handwriting characteristics, languages, and styles that are vulnerable to intrawriter and interwriter differences [9]. Moreover, the handwritten numerals found on electronic documents and images are different in terms of their location, size, space from the file margins, and width, which increase the difficulty of distinguishing them accurately [10]. Due to this diversity, handwritten digit-detection and classification systems are customized for specific applications to improve the overall performance of the system [6, 11].

In the area of HDR, pattern recognition and image processing play an important role in memorizing patterns in handwriting. Recently, several HDR frameworks [12–20] have been presented to recognize handwritten numerals. A typical HDR system involves the following main phases: data preparation, segmentation, key point computation, and categorization [21, 22]. Rapid developments in the area of HDR demonstrate progression in learning algorithms [5, 9, 23] and the accessibility of huge databases [24, 25]. The research community has introduced several solutions utilizing handcrafted key point-based methods [13–16, 26], dense network approaches [18–20, 27], etc. Some of the most often used handcrafted features in character identification include zoning features [28, 29], projections [30], Fourier descriptors [31, 32], contour direction histogram [9, 33], chain coding [33, 34] and invariant moments [35]. These key points can be combined to create a reliable set of features that can be used to train a classifier. The computation of the features can significantly affect a classifier's performance. Numerous classification algorithms, including support vector machines (SVMs) [13–16], k-nearest neighbors (KNNs) [36], decision trees (DT) [37], and random forest (RF) [13] have been adopted. A reliable set of features should represent all characteristics of handwriting that are specific to a particular category and be as discriminatory as possible against all other classes [38]. However, handcrafted feature-based approaches are found ineffective due to the substantial amount of time required for data preparation and increased training time. Moreover, these approaches are unable to effectively locate the numerals from distorted images.

Now, DL is a rapidly expanding field among other machine-learning (ML) approaches for achieving improved performance in the areas of pattern recognition [39, 40] and character detection [23, 41–43] due to its discriminative key point computation and classification properties. Among other frameworks of DL, deep neural networks (DNNs) are more time-consuming due to the increased number of hidden units and links [44]. From the DL family, the CNN approaches are currently more popular for image analysis since these models employ fewer hidden layers than a DNN and have fewer parameters [40]. Moreover, these approaches are proficient to compute location-invariant key points in a viable timeframe because of their effective pattern-recognition capabilities. Furthermore, CNNs are proficient in mapping input information to the output using

temporal subsampling and thus are not affected by distortions or basic geometric modifications such as rotation, translation, squeezing, or scaling operation [45]. Since handwritten numerals can be found in a variety of styles and orientations, therefore, several CNN algorithms have been extensively investigated by researchers to address challenges in the domain of automated HDR [46–48]. Authors [16] suggested a CNN and SVM-based HDR system and showed improved classification results. In [5], another CNN-based approach, namely, the binary convolutional neural network (BCNN), for HDR is proposed. This approach generated high recognition results but was unable to learn the advanced characteristics of input samples. In [47], the authors suggested two designs for HDR using feedforward NN along with the CNN model for key point computation and categorization. The results show that CNN outperforms FWNN when it comes to handwritten numeral identification. In [48], an ensemble technique for HDR composed of various CNN architectures to improve classification efficiency at the expense of greater computational cost is developed. Although these prior studies [5, 47, 48] have attained excellent recognition precision, there is still potential for the development of automated HDR in terms of speed and accuracy performance. Several approaches have been presented in history for the effective and efficient detection of numerals from images, where most of the works are based on DL approaches. The main shortcoming of DL-based HDR techniques is their inefficiency and significant processing times. Moreover, many contributions in the literature for the recognition of handwritten and printed text recognition focus primarily on simple characters with no anomalies or noise [49]. However, in real scenarios, the handwritten characters can be broken/distorted and produced in unrestricted conditions such as noise, blurriness along with varying illumination conditions, contrast, and intensity. These factors cause identification algorithms to exhibit multiple contradicting behaviors, thus affecting overall recognition accuracy. Rani et al. [46] proposed a deep CNN model based on AlexNet for the identification of characters having broken and messy appearances. The accuracy of the model is reported at 92% of the synthetically gathered database. Similarly, in [23], an object recognition framework was adopted for the identification of handwritten numerals under the presence of noise and distortions. This approach attains comparatively higher recognition results but at the charge of enhanced computational burden.

The proposed study varies from previous research [23, 46–48] as it demonstrates the usefulness of CNN in terms of high accuracy as well as computational and processing complexity for categorizing handwritten numerals. The main motivation of this work is to present an efficient and effective solution to the aforementioned issues of numeral recognition. For this, we suggested an efficient CNN framework for HDR that is inspired by the EfficientDet model [50], which produces higher recognition rates in comparison to the existing latest approaches. In the EfficientDet-D4 model, the EfficientNet-B4 CNN backbone aids in the extraction of more reliable information from the input handwritten numeral image. The proposed DL framework

can accurately detect and identify handwritten numbers in the presence of distorted backgrounds because it uses multiscale feature fusion during feature computation. The experimental findings validate the effectiveness of the proposed approach for dealing with complex situations such as changes in size, orientation, writing formats, and styles and huge matching among the structures of various digits. Moreover, the presented framework is also robust against noise and blurring in the input samples and can efficiently recognize numerals by minimizing the execution time due to its shallow architecture.

The following are the main contributions of the proposed technique:

(1) We presented an end-to-end deep learning framework based on the EfficientDet-D4 model for precise identification and classification of handwritten numerals

(2) The presented approach uses the lightweight backbone EfficientNet-B4 to compute robust and discriminative key points that improve the overall performance of HDR while reducing model training and execution time

(3) To exhibit the usefulness of the presented framework, a rigorous quantitative and qualitative comparison of the provided technique was undertaken using a standard benchmark MNIST database. The findings show that the proposed CNN model improves recognition rates when compared to previous CNN-based algorithms

The remaining structure of this study is as follows: Section 2 presents an overview of the related work that has been done to identify handwritten numbers from images. Section 3 provides details of the proposed methodology adopted for the recognition purpose. In Section 4, the experimental setting and findings are discussed, and lastly, the study is concluded with some recommendations in Section 5.

2. Related Work

Significant work has been done in the past for the development of automated HDR systems. Boukharouba and Bennia [9] suggested a handcrafted feature-based approach for HDR. A distinctive key point set was generated by employing the chain code histogram (CCH) [33] approach along with the pixel's transition information in the horizontal and vertical directions. The SVM classifier was then trained to classify extracted key points. This approach [9] is capable of accurately recognizing handwritten numerals; however, it necessitates extensive training on a larger database. In [51], the author proposed an ensemble classification method using bagging for improved accuracy. A hybrid system based on bagged-SVM, bagged-RBF, and RBF-SVM was designed and evaluated on different real and benchmark datasets. The performance evaluations demonstrated that the suggested hybrid RBF-SVM classifier performs well in terms of classification results; however, the generalization power needs further enhancements. Similarly in [52], the authors

introduced a new approach for identifying handwritten digits by combining different feature-extraction methods and employing ensemble classifiers. Six feature sets were obtained in total, and the MNIST database was used to evaluate this model. The results demonstrated improved recognition performance; however, noise, distortion, or an unusual writing style cause performance degradation. Dine et al. [53] presented a novel feature computation method based on structural and statistical approaches. Initially, the preprocessing was performed to binarize, crop, and normalize the input data. Then, four different feature sets were computed by using cavity [54], zoning [28], Freeman chain coding (FFC) [34], and profile projection [55] methods. Lastly, KNN was used for the classification of key points. This approach [53] attained a recognition accuracy of 95% using FFC on the MNIST database. However, it may not perform well on challenging databases. Hou and Zhao [56] employed a combination of both handcrafted and deep features for the recognition of handwritten numerals. The Gabor feature-extraction algorithm was used to compute the handcrafted key points. The CNN classifier was trained using the calculated key points. This method [56] enhanced the accuracy of number identification; however, it has an increased computational cost. Pham et al. [57] suggested a dropout regularization strategy to improve the resilience of RNNs for HDR. This approach increases RNN accuracy by significantly lowering character and word error rates. Shamim et al. [13] developed an offline HDR technique using several ML algorithms such as multilayer perceptron (MLP), Naive-Bayes, Bayes-Net, SVM, J48, RF, and random tree. The results showed that MLP outperforms other classifiers in recognition performance. In [58], the authors presented a convolutional recurrent neural network (CRNN) that combined the advantages of deep CNN with RNN. This technique [58] was also used for scene text categorization and outperformed existing approaches for number recognition; however, this technique is computationally complex.

Wang et al. [59] introduced the quantum k-neighbor technique for HDR. The method in [59] has minimized the economic burden in comparison to the traditional k-nearest neighbor algorithm; however, the performance needs further improvement. Another technique for HDR was proposed in [60]. In the first step, the multizoning technique [61] was applied for the key point calculation. In the next step, SVM and MLP classifiers were trained over the computed features to locate the handwritten digits. The method in [60] exhibits better numeral recognition performance but may not show better detection accuracy when digits form a triangle shape. Assegie and Nair [37] introduced an approach for HDR. Image pixels were employed as feature vectors to train the DT for digit localization and detection. This approach [37] is simple to implement; however, recognition accuracy needs further enhancement. Ali et al. [42] proposed a framework for localizing and recognizing the numerals, in which the Java-based DL4J model was utilized for the key point computation. In the next step, the extracted features were utilized to train CNN to classify the detected digits. It is concluded in [42] that for small-sized databases, CNN with minimum layers exhibits better detection

accuracy. Another DL-based framework was introduced in [62] to automatically locate the digits from input samples named as the deep convolutional self-organizing map (DCSOM) framework. The method explained in [62] employed the DCSOM model to compute the local histograms and deep key points to exhibit the categorization results. This method [62] works well for digit categorization under the occurrence of noise but may not perform well under rotational variations. Hafiz and Bhat [18] proposed an effective hybrid classifier by applying DL with the Q-learning-based reinforcement learning method [63]. This work [18] shows better digit categorization accuracy under the occurrence of rotational alterations but at the expense of increased economical burden. A 3-layered spiking neural network (SNN) for recognizing the numerals was proposed in [19]. It is concluded in [19] that the SNN-based DL model works well for numeral classification in comparison to traditional ANN frameworks with backpropagation (BP) techniques. However, such solutions may not perform well over large benchmark classification problems. Other researchers modified the BP algorithm by adopting the variable step size approach and the Newton method to improve the network convergence speed [27]. Although this methodology enhanced the algorithm's handwritten numeral recognition performance, it requires more memory and is therefore inadequate for large-scale applications. In [20], a rapid handwritten numeral identification technique was proposed based on affinity propagation (AP) clustering and the BP neural network. However, the aforementioned methods [20, 27] lack the ability to represent robust features and are quickly influenced by external factors such as noise and blurriness, making them unable to achieve better recognition accuracy. Moreover, the described studies [20, 27] include limitations in terms of accuracy and computing time that can be further improved.

Jain et al. [64] presented a rotational invariant architecture using the CNN model for the identification of handwritten digits and captcha recognition. The approach employs multiple instances of the LeNet model [4]; each model is trained on different rotation angles and is, thus, robust to high-degree variation in the orientation of digits. Ali et al. [41] introduced a CNN with an extreme learning machine- (ELM-) based method for the recognition of handwritten numbers. For the extraction of discriminative key points, an enhanced CNN based on the LeNet model [4] having 5 hidden layers and an ELM classifier for key point classification into 0-9 classes was built. This strategy [41] improves the classification accuracy; however, the model suffers from the overfitting problem. The authors also investigated the effect of increasing hidden layers on model performance and discovered that increasing hidden layers reduces model performance for HDR. Similarly, in [65], the authors suggested an ANN and ELM-based technique for the classification of MNIST handwritten digits. The results showed that ELM attains better classification accuracy and less processing time than the ANN approach. Albahli et al. [23] proposed a region-based CNN, namely, Faster-RCNN, for the HDR system. The authors employed an improved key point extraction network based on DenseNet-41 for the computation discriminative feature

set. The region proposal network was then used to produce the ROI and classify the handwritten numerals. This method [23] correctly recognized digits under complex transformations including rotation and scale variations; however, it relies on a predetermined set of anchor boxes and thus involves extensive hyperparameter choices during the training process. Table 1 presents the comparison of existing approaches and their limitations. We reviewed numerous automated approaches for accurately distinguishing handwritten numbers; nevertheless, performance can still be further improved in terms of accuracy and computing complexity. Existing methods either necessitate extensive preprocessing or are ineffective when dealing with distorted backgrounds or complex situations such as changes in size, orientation, writing formats, styles, noise, and blurriness. Furthermore, these methods require considerable training and are susceptible to model overfitting, resulting in poor performance on unseen data that can be enhanced.

3. Method and Material

To design the numeral recognition system, we have followed two main phases: (i) preparing data and (ii) handwritten digit localization and classification. The entire flow of the introduced approach is explained in Figure 1 while the mathematical model formulation is shown in Algorithm 1. In the first phase, annotations are designed to exactly locate the area containing each numeral. Next, the created annotations are used to perform the training of the introduced approach, namely, the EfficientNet-B4-based EfficientDet-D4. The EfficientDet-D4 model consists of three stages to perform the detection and categorization of HDR. Initially, EfficientNet-B4 is used as the base network to calculate the features from the input image. The EfficientDet-D4 framework accepts two types of inputs, namely, the input and annotated samples. Then, the BiFPN unit of the EfficientDet-D4 approach executed the top-down and bottom-up feature concatenation numerous times to calculate the final feature vector. Next, the prediction module performs the localization and classification of numerals via employing the computed key points, and obtained performance results are determined by using the standard metrics employed in the field of deep learning. A detailed demonstration of phases followed by the introduced work is explained in Algorithm 1, while the in-depth details of all steps are discussed in the subsequent sections.

3.1. Dataset. To analyze the classification accuracy of the presented framework, several experiments are performed over the challenging database, namely, the MNIST (Modified National Institute of Standards and Technology) dataset [24]. This dataset contains 10 classes of numerals presenting digits from 0 to 9. MNIST is a standard and the largest dataset of handwritten digits that is also heavily explored by the research community to train numerous image-processing models. The used dataset consists of a total of 70,000 samples of which 60,000 images are from the training part, while the remaining 10,000 samples are from the test set. The images from the MNIST dataset are

TABLE 1: Comparison of existing approaches of handwritten digit recognition.

Reference	Method	Dataset	Results	Limitations
Boukharouba and Bennia [9]	Handcrafted features using chain code histogram with SVM classifier	Handwritten Farsi digit dataset, i.e., Hoda	Accuracy = 98.55	Unable to distinguish between similar classes
Govindarajan [51]	Ensemble classifier using bagged-SVM, bagged-RBF, and RBF-SVM	MNIST dataset	Accuracy = 98.0%	Lack of robustness to unseen samples
Cruz et al. [52]	Gradient directional features with MLP classifier	MNIST dataset	Accuracy = 99.68%	Performance degrades on distorted samples
Dine et al. [53]	Structural and statistical features with KNN classifier	MNIST dataset	Accuracy = 95%	Computationally inefficient due to large vector size
Hou and Zhao [56]	Hybrid feature using Gabor filter and CNN with ANN classifier	MNIST dataset	Accuracy = 99.23%	Performance degrades for samples with intense rotation variation
Pham et al. [57]	RNN	Handwriting datasets, i.e., RIMES, IAM, and OpenHaRT	Character error rate = 9.17% (best on OpenHaRT)	Needs performance improvement
Shamim et al. [13]	Compared several ML algorithms such as MLP, Naive-Bayes, Bayes-Net, SVM, J48, RF, and random tree	Digits-A dataset	Accuracy = 90.37%	Needs performance evaluation on other complex datasets as well
Shi et al. [58]	CRNN based on CNN with RNN	Datasets such as IIIT-5k, Street View Text, and ICDAR	Accuracy = 98.7%	Limited for practical applications due to high inference time
Arbian et al. [60]	Handcrafted features using multizoning technique and SVM and MLP classifier	MNIST dataset	Accuracy = 95.35%	Model performance degrades on shape and size variations
Assegie and Nair [37]	Spatial features with decision tree classifier	MNIST dataset	Accuracy = 83.4%	Performance needs improvement
Ali et al. [41]	CNN with ELM classifier	MNIST dataset	Accuracy = 99.8%	Requires performance generalization on unseen data
Aly and Almotairi [62]	Deep convolutional self-organizing map	MNIST dataset	Error rate = 0.57%	Performance degrades on geometrical variability such as rotation
Jain et al. [64]	Modified LeNet-CNN	MNIST dataset	Accuracy = 99.5%	Low performance on intense rotational variations
Malik and Roy [65]	Spatial and spectral features with ANN and ELM classifier	MNIST dataset	Accuracy = 98.4% (ELM)	Performance needs improvement
Albahli et al. [23]	DenseNet-faster RCNN	MNIST dataset	Accuracy = 99.78%	The method is computationally expensive

challenging in nature as these are subject to variations in the size, scale, and angle of digits. Moreover, the images are suffering from blurring, noise, and intensity variations as well which present it as a complex database for numeral classification. We have presented some sample images from the MNIST dataset in Figure 2.

3.2. Annotations. For the training procedure, it is necessary to correctly determine the location of digits from the input images. To accomplish this, the labeling [26] program is used to make the annotations of digit image portions to precisely indicate the ROIs. The generated values are stored in XML files having Bbox values along with the class of the region. After that, these files are used to generate the training file which is required for framework training.

3.3. EfficientDet. Accurate and efficient feature extraction is essential to classify the input samples as digit categories. Simultaneously, getting the more descriptive feature set is a difficult task due to some reasons like the calculation of a
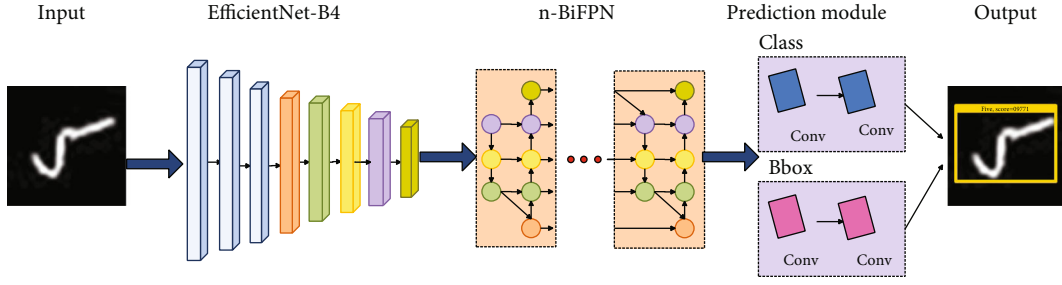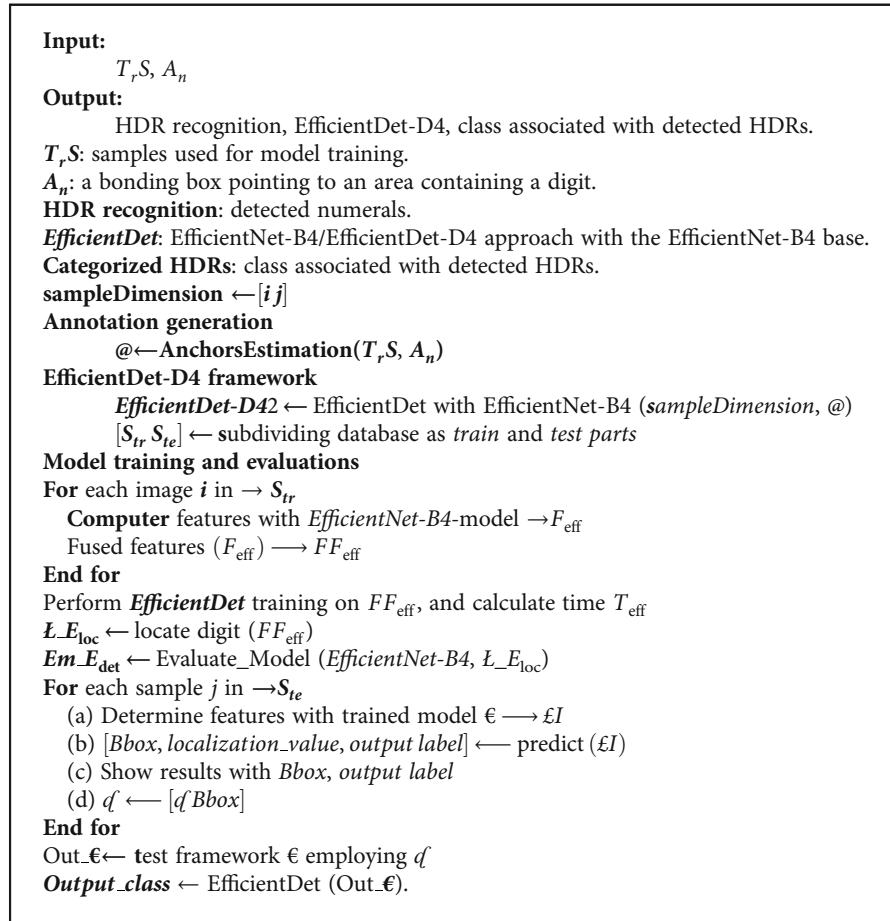
FIGURE 1: Flow diagram of the proposed technique.

Input:
T_rS, A_n
Output:
HDR recognition, EfficientDet-D4, class associated with detected HDRs.
T_rS: samples used for model training.
A_n: a bonding box pointing to an area containing a digit.
HDR recognition: detected numerals.
EfficientDet: EfficientNet-B4/EfficientDet-D4 approach with the EfficientNet-B4 base.
Categorized HDRs: class associated with detected HDRs.
sampleDimension $\leftarrow [i\,j]$
Annotation generation
@$\leftarrow$**AnchorsEstimation**(T_rS, A_n)
EfficientDet-D4 framework
EfficientDet-D42 $\leftarrow$ EfficientDet with EfficientNet-B4 (**sampleDimension**, @)
$[S_{tr}, S_{te}] \leftarrow$ subdividing database as *train* and *test parts*
Model training and evaluations
For each image i in $\rightarrow S_{tr}$
Computer features with *EfficientNet-B4*-model $\rightarrow F_{eff}$
Fused features $(F_{eff}) \longrightarrow FF_{eff}$
End for
Perform **EfficientDet** training on FF_{eff}, and calculate time T_{eff}
$\textit{Ł\_E}_{loc} \leftarrow$ locate digit (FF_{eff})
$\textit{Em\_E}_{det} \leftarrow$ Evaluate_Model (*EfficientNet-B4*, $\textit{Ł\_E}_{loc}$)
For each sample j in $\rightarrow S_{te}$
(a) Determine features with trained model € $\longrightarrow £I$
(b) $[Bbox, localization\_value, output\ label] \longleftarrow$ predict $(£I)$
(c) Show results with *Bbox, output label*
(d) $q \longleftarrow [q\,Bbox]$
End for
Out_€$\leftarrow$ **t**est framework € employing q
Output_class $\leftarrow$ EfficientDet (Out_€).

ALGORITHM 1: HDR classification using the EfficientDet-D4 model.

large feature set which causes the model overfitting, and the other one is a smaller-size feature set which causes the model to miss learning some important image characteristics, i.e., size, color, and texture. So, it is essential to utilize an automated feature extractor instead of handcrafted feature-extraction methods which can estimate the more representative features from the input images. The handcrafted feature-estimation methods are not effective in precisely localizing and categorizing the digits due to different factors, i.e., position, shape, color, chrominance, etc. To overcome these challenges, we have used the EfficientDet DL method which can extract features from images under examination.

The convolution filters compute the features of the image by exploring the structure of an object. Numerous object-detection techniques have been introduced for the recognition of several diseases. These methods are divided into one-stage (YOLO [66], CornerNet [67], CenterNet [68], etc.) and two-stage object detectors (Fast-RCNN [69], Faster-RCNN [70], Mask-RCNN [39], EfficientDet [71], etc.). The reason for choosing EfficientDet over one-stage methods is that these approaches compromise the detection accuracy by exposing the least time to perform classification, whereas the two-stage methods show better digit-detection performance; however, these approaches are computationally complex due to

FIGURE 2: Visual representation of samples from the MNIST dataset.

their two-stage network architecture which makes them inappropriate for real-world examples. So, there is a demand to present a method that will provide an efficient solution to digit classification and recognition of handwritten images.

To tackle these problems, we have employed the EfficientDet technique (proposed by the Google Brain team). The EfficientDet method is a more robust and scalable model due to the feature pyramid network (FPN) architecture which calculates the multidirected feature fusion. The proposed method has three main components, i.e., feature estimation through EfficientNet-B4, bidirectional FPN (BiFPN) used for both top-down and bottom-up feature fusion, and the final component which is utilized to localize and classify digits.

The following are the three modules of our method.

3.3.1. Feature Extraction through EfficientNet-B4. For the extraction of features, we have utilized the approach, namely, EfficientNet-B4 as a backbone network of the EfficientDet technique. The EfficientNet technique reliably balances each dimension with a static set of scaling coefficients as compared to the traditional models which perform scaling randomly. The employed feature extractor, namely, the EfficientNet-B4, is empowered to calculate a discriminative set of sample key points while maintaining the minimum set of framework parameters that assist to enhance the recognition performance of the proposed solution. The EfficientNet approach can easily handle the various transformations of the input images which allows it to robustly tackle the problem of the nonexistence of the digit's position information. Moreover, the EfficientNet model permits the reemployment of the extracted key points which enables it to be more appropriate for HDR detection and classification and speed up the training method as well. An illustration of the feature extractor EfficientNet-B4 is presented in Figure 3.

3.3.2. BiFPN. For the efficient detection and classification of HDR, several object transformations like the location, shape, structure of digits, sample background, and intensity variations must be taken into count. Hence, employing multiscale feature extraction can play a vital role in correctly identifying the digits. Typically, the traditional models only utilize the top-down FPNs to combine the multistage features which lacks being able to completely handle the scale variations of samples and causes missing to learn the important aspects of object structure information. Therefore, such

architectures of models are not well generalized to recognize the digits of varying size and with different angle orientations which ultimately cause degradation of the numeral detection and classification performance. To deal with the above-mentioned problems, the proposed approach gives the idea of BiFPN which causes the information to move in both the top-down and bottom-up directions by utilizing consistent and well-organized links. Furthermore, the BiFPN unit exploits trainable weights to calculate semantic features which ultimately enhances the HDR performance of the introduced framework. Hence, such a structure of the framework allows selection of the more important set of image features by the EfficientNet-B4 model which is used as input by the BiFPN unit. The depth and width of BiFPN are obtained, mathematically defined in the following equation:

$$w_b = 64 \cdot \left(1.35^{\varnothing}\right), d_b = 3 + \varnothing. \tag{1}$$

Here, w_b and d_b are depicting the width and depth of the BiFPN unit, respectively, whereas $\varnothing$ which is the compound element controls the scale sizes and has a value of 0 for the proposed approach.

3.3.3. Box/Class Prediction Module. The resultant features are given to the box/class prediction network to find out the Bbox values towards the digit areas along with the respective class. The thickness of the current network is the same as BiFPN; however, depth is calculated as follows:

$$D_{\text{Bbox}} = 3 + \left\lceil \frac{\varnothing}{3} \right\rceil. \tag{2}$$

3.4. Detection Procedure. The proposed technique is free from other approaches, i.e., proposal generation and selective search. Moreover, the input along with the respective annotations is put into the EfficientDet, which immediately calculates the digit location and its respective class.

4. Experimental Results

In this section, we have provided a detailed description of the employed dataset along with the metrics which are used to assess the performance of the proposed model. Moreover, we have performed a series of experiments to check the numeral detection and classification performance of the presented approach. We have implemented the proposed method in Python language and executed it on an Nvidia GTX1070 GPU-based system. Table 2 displays the details of the training parameters of the proposed work. We have reported training and loss curves in Figure 4 to show the optimized learning behavior of the proposed approach.

4.1. Evaluation Metrics. To evaluate the numeral identification and categorization performance of the proposed approach, we have employed numerous standard performance measurements named intersection over union (IOU), accuracy,
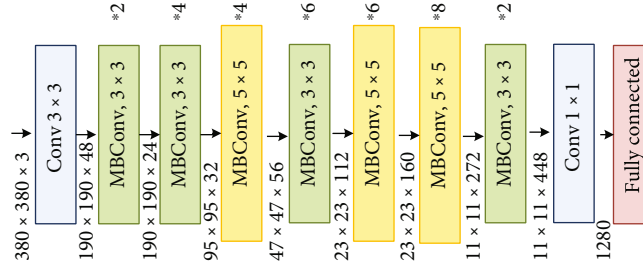
FIGURE 3: Architecture of feature extractor EfficientNet-B4.

TABLE 2: Training parameters of the proposed solution.

Framework parameters	Value
Epochs	25
Learning rate	0.01
Batch size	90
Confidence score threshold	0.5
Unmatched threshold	0.5

precision, recall, and mean average precision (mAP). The framework classification performance is calculated by using the following equation:

$$mAP := \sum_{i=1}^{T} \frac{AP(t_i)}{T}. \qquad (3)$$

Equation (3) exhibits the computation of the mAP score, in which AP is depicting the average precision value which is calculated for all categories, whereas t is indicating the image under evaluation. Furthermore, T symbolizes the total number of samples. Figure 5 displays the visual depiction of IOU, precision, and recall.

4.2. Model Evaluation. To check the numeral recognition and categorization results of the presented approach, we have performed two types of evaluations, namely, the digit localization results and the class-wise performance. This analysis will assist the reader to determine the HDR localization and classification performance.

4.2.1. Numeral Localization Results. An accurate HDR system must be enabled to correctly identify the ROIs (numbers) of different types. To check this, we have conducted an analysis. We have taken all the test images of the MNIST dataset and evaluated them on the trained EfficientDet-D4 model, and a few samples are shown in Figure 6. The localization results shown in Figure 6 clearly explain that the proposed approach is empowered to detect the digits of several types efficiently. Moreover, the technique is robust with regard to several image distortions with the existence of noise, blurring, and intensity alterations and can locate the digits with the size, angle, and position variations. We have also quantitatively measured the localization power of the employed framework by using two standard metrics, namely, the mAP and IOU scores. More clearly, the EfficientDet has classified the HDR with the mAP and

IOU scores of 0.995 and 0.994, respectively, which demonstrates the effectiveness of the proposed solution.

4.2.2. Class-Wise Results. To design the efficient numeral classification approach, it must be capable of differentiating the digits of different classes. Therefore, an analysis is performed to validate the presented technique for class-wise classification performance. Several experiments have been conducted to demonstrate the classification performance of the introduced work. Initially, we computed the class-wise precision, recall, F1-score, and error rate, and the attained results are explained in Table 3. The values demonstrated in Table 3 clearly show that our work is robust towards numeral recognition. More clearly, the introduced approach has acquired the average precision, recall, and F1-score of 98.75, 98.2%, and 98.45%, respectively, along with an average error rate of 1.55%.

To further discuss the numeral categorization performance of the EfficientDet-D4 model, we have discussed the class-wise accuracy values of the introduced method with the help of the bar graphs as these provide better visualization of data (Figure 7). More clearly, our approach has achieved the average accuracy values of 99.71%, 99.73%, 99.84%, 99.79%, 99.91%, 99.87%, 99.85%, 99.85%, 99.89%, and 99.85% for classes from zero to nine, respectively, that show the robustness of the introduced method. The obtained performance values clearly show that our method has exhibited efficient accuracy results for all classes of digits.

Moreover, we have plotted the confusion matrix for the proposed approach, as it assists in determining the recall rate of the model and its capability to recognize the digits of several classes. The acquired values are explained in Figure 8 which clearly shows the accurateness of the introduced methodology for the numeral classification. More descriptively, we have obtained the TPR rates of 0.960, 0.970, 0.980, 0.990, 0.983, 0.992, 0.992, 0.988, 0.988, and 0.983 for numerals from zero to nine, respectively. It is quite visible from Figure 8 that the proposed model can correctly recognize and classify the digits, even though a little similarity exists among digit 1 and 7 classes; however, both are still distinguishable.

It can be summarized based on the evaluation results of the above-performed analysis that the EfficientDet-D4 model is robust for detecting and classifying the digits in their respective classes. The major cause for the better classification results of the introduced technique is because of the representative key point extraction ability of the
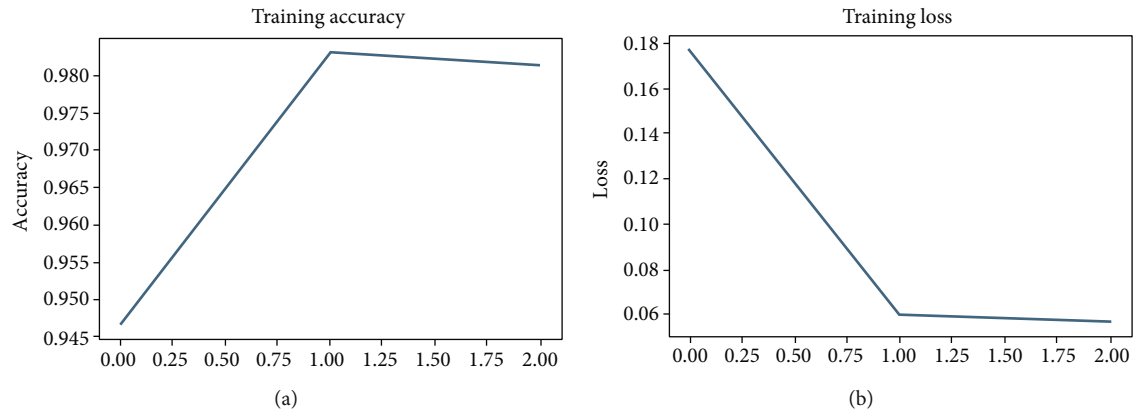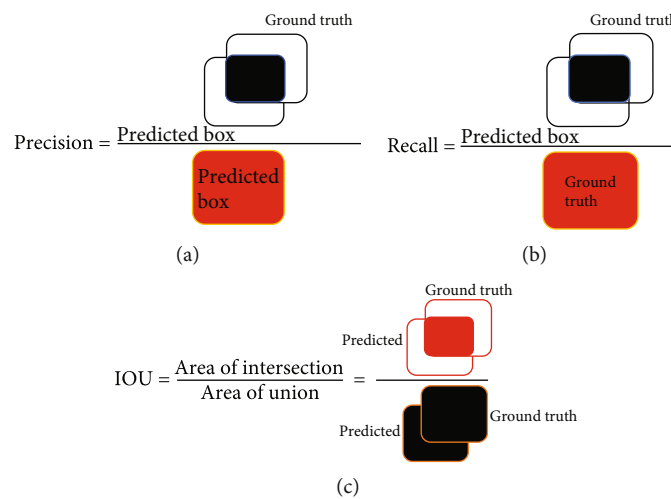
FIGURE 4: Training and loss curves.

FIGURE 5: Visual demonstration of precision, recall, and IOU metrics.

EfficientDet-D4 which permits it to better recognize the numerals. Furthermore, the EfficientDet-D4 approach can easily deal with the overtuned model data and handle the complex image transformations which result in its effective performance.

4.3. Comparative Analysis with DL-Based Methods. We have designed an experiment to evaluate the introduced method against several DL-based approaches. To accomplish this, we have chosen several methods, namely, RCNN, Fast-RCNN, Faster-RCNN, SSD, and YOLO. To conduct the performance analysis, we have taken the mAP and accuracy evaluation metrics, as these are the standard measures employed by the researchers to compute the classification results of object-detection methods. Moreover, we have compared the execution times of all methods to show the effectiveness of our model. The comparative results are demonstrated in Table 4, from where it can be seen that our approach outperforms the rest of the methods with the mAP score, accuracy, and time values of 0.995, 99.83%, and 0.16 s, respectively. The Faster-RCNN approach shows

the comparative results with the mAP and accuracy values of 0.993, and 99.78%, respectively, while the lowest performance is exhibited by the YOLO framework with the mAP and accuracy values of 0.943 and 93.20%, respectively. More descriptively, in the terms of mAP evaluation measurement, the competitive methods give an average value of 0.961, which is 0.995 for our method. So, for the mAP, the proposed approach gives a performance gain of 3.4%. Similarly, for the accuracy measure, the competent methods present an average accuracy measure of 96.29%, while in comparison, the presented work shows an average accuracy value of 99.83% and hence gives a performance gain of 3.54%. Furthermore, we have compared the execution time as well, and it is quite evident that our method shows the lowest execution time of 0.16 s as compared to all other approaches. So, it can be concluded that the EfficientDet approach gives a reliable and low-cost solution to numeral identification and classification. The basic cause for the better results of the introduced method is because of the robust key point extraction ability of the EfficientDet-D4 model which presented the complicated image transformations in a viable manner.

FIGURE 6: HDR localization results obtained with the proposed technique.

TABLE 3: Class-wise performance results of the EfficientDet-D4 for HDR recognition.

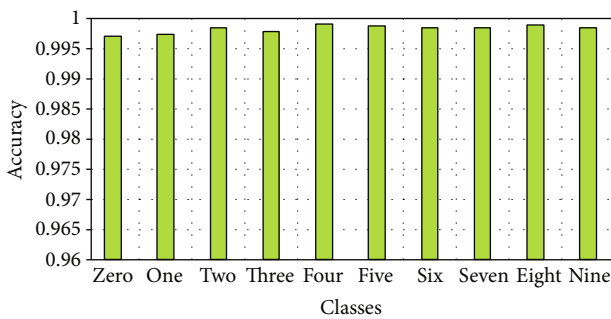| Classes	Precision	Recall	F1-score	Error rate
Zero	0.999	0.960	0.979	0.021
One	0.988	0.970	0.963	0.037
Two	0.987	0.980	0.994	0.006
Three	0.971	0.990	0.980	0.020
Four	0.972	0.983	0.976	0.024
Five	0.987	0.992	0.989	0.011
Six	0.999	0.992	0.996	0.004
Seven	0.990	0.988	0.989	0.011
Eight	0.983	0.988	0.990	0.01
Nine	0.999	0.983	0.989	0.011

FIGURE 7: Class-wise accuracy values.

Moreover, the one-stage digit identification and classification ability of the EfficientDet-D4 approach provides it with a computational advantage as well.

4.4. Comparative Analysis with ML-Based Methods. To discuss the efficiency of the introduced method, another experiment has been performed to check the HDR results of the EfficientDet approach against the ML-based classifier. For this reason, we have taken three renowned ML-based classifiers, namely, the KNN, DT, and SVM, and compared their results against our approach as demonstrated in [72, 73]. The obtained results are shown in Table 5 from where it is quite clear that our method exhibits the highest classification performance with an accuracy value of 99.83%. The KNN method shows the second-highest classification performance with 96.89% accuracy. Moreover, the lowest classification performance is shown by the DT classifier with an accuracy value of 86.60%. More clearly, the methods in [72, 73] acquire an average accuracy value of 92.42%, while it is 99.83% for the proposed method. Hence, we have demonstrated an average performance gain of 7.41%. The reported results are clearly showing that the proposed solution exhibits better numeral detection accuracy in comparison to the ML classifiers because of its power to tackle the overtuned framework training data.

4.5. Comparative Analysis with State of the Art. In this part, We have performed another experiment where we have taken numerous latest methods from the literature which use the same database for numeral identification and categorization and compared our results against them. To have an

True class	Zero	One	Two	Three	Four	Five	Six	Seven	Eight	Nine
Zero	96%	0.2%	1%	0.2%	0.5%	0.1%	0.5%	0%	0.5%	1%
One	1%	97%	0%	0.5%	0%	0%	0%	1%	0.5%	0%
Two	1%	0%	98%	0.1%	0.4%	0.2%	0%	0%	0.1%	0.2%
Three	0%	0%	0.6%	99%	0.2%	0%	0%	0%	0%	0.2%
Four	0.5%	0.5%	0.2%	0%	98.3%	0.1%	0.3%	0%	0.1%	0%
Five	0%	0.8%	0%	0%	0%	99.2%	0%	0%	0%	0%
Six	0.5%	0.3%	0%	0%	0%	0%	99.2%	0%	0%	0%
Seven	0%	0%	0%	0.2%	0.5%	0.2%	0%	98.8%	0%	0.3%
Eight	0%	1%	0.2%	0%	0%	0%	0%	0%	98.8%	0%
Nine	1%	0.2%	0%	0%	0.1%	0.2%	0%	0.2%	0%	98.3%

Predicted class

Figure 8: Confusion matrix of the proposed approach for HDR.

Table 4: Comparative analysis with DL methods.

Model	mAP	Accuracy	Time
RCNN	0.947	96.19%	0.25
Fast-RCNN	0.966	97.85%	0.22
Faster-RCNN	0.993	99.78%	0.19
YOLO	0.943	93.20%	0.18
SSD	0.958	94.44%	0.19
Proposed	0.995	99.83%	0.16

Table 5: Performance evaluation with ML-based methods.

Method	Accuracy
Deep features (EfficienNet-B4)+KNN	96.89%
Deep features (EfficienNet-B4)+DT	86.60%
Deep features (EfficienNet-B4)+SVM	93.77%
Proposed	99.83%

Table 6: Performance evaluation of the proposed technique with state-of-the-art techniques.

Approaches	Accuracy
Zhao and Liu [74]	98.1%
Enriquez et al. [75]	98%
Ge et al. [76]	95.7%
Beikmohammadi and Zahabi [77]	99.80%
Agrawal [78]	99.60%
Proposed technique	99.83%

unbiased assessment, the average results of the proposed method are evaluated against the average performance values reported in these methods [74–78], and the obtained comparison is given in Table 6. Zhao and Liu [74] introduced a DL-based method named LeNet-5 to categorize the handwritten numerals from the input images and obtained a 98.1% accuracy value. Enriquez et al. [75] introduced a CNN approach for numeral recognition and achieved 98% accuracy results. Similarly, in [76], a DL-based approach for HDR shows an average accuracy value of 95.7%. Beikmohammadi and Zahabi [77] proposed a hybrid CNN model to locate and categorize the digits with an accuracy of 99.80%. Similarly, a DL-based approach was used in [78] for numeral categorization and obtained an average accuracy value of 99.60%, whereas comparatively, the employed EfficientDet-D4 method has gained a 99.83%

accuracy value which is the largest in comparison to all evaluated methods. In the more in-depth analysis, the comparative approaches showed an average accuracy value of 98.24% while it is 99.83% for our technique. Hence, the proposed method gives a performance gain of 1.59%. The basic cause for the robust numeral classification results of the introduced framework is that the works presented in [74–78] use an intense deep-framework architecture for key point calculation, which results in model overfitting and enhancing the computational burden as well, while the presented method uses the EfficientNet-B4 as a base network, which is capable of computing a reliable set of sample key points and preserving the computational complexity as well. Therefore, we can say that the EfficientNet-B4-based EfficientDet-D4 framework gives a robust and effective solution to digit classification.

4.6. Cross-Dataset Evaluation. For a robust HDR system, it must be capable of detecting and classifying the digits from unseen examples. To test the generalization power of the proposed technique, we have performed an evaluation. To accomplish this, we have taken another dataset, namely, the USPS [25]. The USPS dataset consists of 7291 training images along with 2007 test samples containing numerals
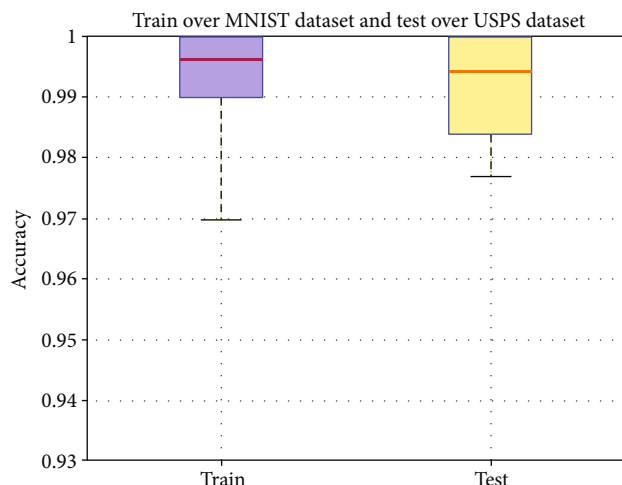
Figure 9: Cross-dataset evaluation results of the proposed technique.

from 0 to 9. We have trained the proposed approach over the MNIST database while the USPS dataset is employed to test the technique. The gained performance values are explained via drawing a box plot (Figure 9), where the performance results for both the training and the test part are ranged over the number line into quartiles, median, and outliers. More clearly, we have obtained the average training and test accuracies of 99.30% and 99.10%, respectively, which demonstrate the robustness of the proposed technique in dealing with unseen examples as well.

5. Conclusion

Accurate recognition of numerals from images plays a significant role in the domain of information processing. However, a huge writing pattern difference and the presence of various sample distortions like noise, blurring, and intensity changes complicate the effective detection of HDR. In this work, a reliable DL-based HDR system, namely, Efficient-Det-D4, is presented to resolve the existing issues of this domain. More clearly, input images are initially annotated to locate the position of digits on images, which are later used to train the EfficientDet model to detect and categorize the digits. We have evaluated the presented approach over the complex dataset, namely, MNIST, and attained an average accuracy value of 99.83%. We have confirmed through huge experimentation that the presented work can efficiently recognize the numerals from the test samples and categorize them into 10 categories showing numbers from 0 to 9. Moreover, the approach is capable of accurately identifying and classifying the digits even under the occurrence of various postprocessing attacks, i.e., light and color variations, blurring, noise, angle and size changes, etc. Furthermore, across-dataset evaluation on the USPS dataset is also accomplished to show the efficacy of the proposed method for the unseen cases. Evaluation results have assured that the introduced approach is robust against present modern techniques and can play a vital role in the area of information processing. Based on the computed results, we can say that this

approach can play an important role in the area of automated number plate recognition of vehicles for surveillance applications. Furthermore, this work has an application in optical character recognition to facilitate various daily life tasks, i.e., product prices, receipt recognition, etc. In the future work, we plan to extend the proposed approach to be applied to other languages.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Disclosure

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

All the authors contributed equally to this research work.

Acknowledgments

References

[1] N. Singh, M. Kumar, B. Singh, and J. Singh, "DeepSpacy-NER: an efficient deep learning model for named entity recognition for Punjabi language," *Evolving Systems*, pp. 1–11, 2022.

[2] M. Kumar, M. Jindal, and M. Kumar, "Distortion, rotation and scale invariant recognition of hollow Hindi characters," *Sādhanā*, vol. 47, no. 2, pp. 1–6, 2022.

[3] M. M. Misgar, F. Mushtaq, S. S. Khurana, and M. Kumar, "Recognition of offline handwritten Urdu characters using RNN and LSTM models," *Multimedia Tools and Applications*, vol. 82, no. 2, pp. 2053–2076, 2023.

[4] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[5] E. Al-wajih, R. Ghazali, and Y. M. M. Hassim, "Residual neural network vs local binary convolutional neural networks for bilingual handwritten digit recognition," in *Recent Advances on Soft Computing and Data Mining. SCDM 2020. Advances in Intelligent Systems and Computing*, R. Ghazali, N. Nawi, M. Deris, and J. Abawajy, Eds., Springer, Cham, 2020.

[6] E. Rashedi and H. Nezamabadi-Pour, "A hierarchical algorithm for vehicle license plate localization," *Multimedia Tools and Applications*, vol. 77, no. 2, pp. 2771–2790, 2018.

[7] A. H. Toselli, V. Romero, M. Pastor, and E. Vidal, "Multi-modal interactive transcription of text images," *Pattern Recognition*, vol. 43, no. 5, pp. 1814–1825, 2010.

[8] M. Busta, L. Neumann, and J. Matas, "Deep textspotter: an end-to-end trainable scene text localization and recognition framework," in *Proceedings of the IEEE international conference on computer vision*, pp. 2204–2212, Venice, Italy, 2017.

[9] A. Boukharouba and A. Bennia, "Novel feature extraction technique for the recognition of handwritten digits," *Applied Computing Informatics*, vol. 13, no. 1, pp. 19–26, 2017.

[10] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[11] B. Epshtein, E. Ofek, and Y. Wexler, "Detecting text in natural scenes with stroke width transform," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2963–2970, San Francisco, CA, USA, 2010.

[12] M. B. Abdulrazzaq and J. N. Saeed, "A comparison of three classification algorithms for handwritten digit recognition," in *2019 International Conference on Advanced Science and Engineering (ICOASE)*, pp. 58–63, Zakho-Duhok, Iraq, 2019.

[13] S. Shamim, M. B. A. Miah, M. R. Angona Sarker, and A. Al Jobair, "Handwritten digit recognition using machine learning algorithms," *Global Journal Of Computer Science Technology*, vol. 3, no. 1, p. 29, 2018.

[14] R. Ebrahimzadeh and M. Jampour, "Efficient handwritten digit recognition based on histogram of oriented gradients and SVM," *International Journal of Computer Applications*, vol. 104, no. 9, 2014.

[15] S. Ahlawat and A. Choudhary, "Hybrid CNN-SVM classifier for handwritten digit recognition," *Procedia Computer Science*, vol. 167, pp. 2554–2560, 2020.

[16] X. X. Niu and C. Y. Suen, "A novel hybrid CNN-SVM classifier for recognizing handwritten digits," *Pattern Recognition*, vol. 45, no. 4, pp. 1318–1325, 2012.

[17] A. Goltsev and V. Gritsenko, "Investigation of efficient features for image recognition by neural networks," *Neural Networks*, vol. 28, pp. 15–23, 2012.

[18] A. M. Hafiz and G. M. Bhat, "Reinforcement learning based handwritten digit recognition with two-state Q-learning," 2020, http://arxiv.org/abs/2007.01193.

[19] S. R. Kulkarni and B. Rajendran, "Spiking neural networks for handwritten digit recognition–supervised learning and network optimization," *Neural Networks*, vol. 103, pp. 118–127, 2018.

[20] Z. Tingting, W. Haikun, and Z. Kanjian, *Handwritten digit recognition based on AP and BP neural network algorithm*, China Sciencepaper, 2014.

[21] L. Neumann and J. Matas, "A method for text localization and recognition in real-world images," in *Asian conference on computer vision*, pp. 770–783, Springer, 2010.

[22] K. S. Younis and A. A. Alkhateeb, "A new implementation of deep neural networks for optical character recognition and face recognition," *Proceedings of the New Trends in Information Technology*, , pp. 157–162, The University of Jordan, 2017.

[23] S. Albahli, M. Nawaz, A. Javed, and A. Irtaza, "An improved faster-RCNN model for handwritten character recognition," *Arabian Journal for Science and Engineering*, vol. 46, no. 9, pp. 8509–8523, 2021.

[24] L. Deng, "The MNIST database of handwritten digit images for machine learning research [best of the web]," *IEEE Signal Processing Magazine*, vol. 29, no. 6, pp. 141-142, 2012.

[25] S. Maji and J. Malik, "Fast and accurate digit classification," EECS Department, University of California, Tech. Rep. UCB/EECS-159, Berkeley, 2009.

[26] F. Lauer, C. Y. Suen, and G. Bloch, "A trainable feature extractor for handwritten digit recognition," *Pattern Recognition*, vol. 40, no. 6, pp. 1816–1824, 2007.

[27] X. Wang, T.-l. Huang, and X.-y. Liu, "Handwritten character recognition based on BP neural network," in *2009 Third International Conference on Genetic and Evolutionary Computing*, pp. 520–524, Guilin, China, 2009.

[28] D. Impedovo and G. Pirlo, "Zoning methods for handwritten character recognition: a survey," *Pattern Recognition*, vol. 47, no. 3, pp. 969–981, 2014.

[29] G. Pirlo and D. Impedovo, "Adaptive membership functions for handwritten character recognition by Voronoi-based image zoning," *IEEE Transactions on Image Processing*, vol. 21, no. 9, pp. 3827–3837, 2012.

[30] A. L. Koerich, *Unconstrained handwritten character recognition using different classification strategies*, International Workshop on Artificial Neural Networks in Pattern Recognition (ANNPR), 2003.

[31] H. Kauppinen, T. Seppanen, and M. Pietikainen, "An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 2, pp. 201–207, 1995.

[32] Y. Lu, S. Schlosser, and M. Janeczko, "Fourier descriptors and handwritten digit recognition," *Machine Vision and Applications*, vol. 6, no. 1, pp. 25–34, 1993.

[33] J. Iivarinen and A. J. Visa, "Shape recognition of irregular objects," in *Proceedings Volume 2904, Intelligent Robots and Computer Vision XV: Algorithms, Techniques,Active Vision, and Materials Handling*, pp. 25–32, Boston, MA, United States, 1996.

[34] F. Groen and P. Verbeek, "Freeman-code probabilities of object boundary quantized contours," *Computer Graphics and Image Processing*, vol. 7, no. 3, pp. 391–402, 1978.

[35] C.-H. Teh and R. T. Chin, "On image analysis by the methods of moments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, no. 4, pp. 496–513, 1988.

[36] Y. Wang, Q. Liu, Y. Sun, F. Zhang, and Y. Zhu, "Design and implementation of handwritten digit recognition based on K-nearest neighbor algorithm," in *The 10th International Conference on Computer Engineering and Networks. CENet 2020. Advances in Intelligent Systems and Computing, vol 1274*, Q. Liu, X. Liu, T. Shen, and X. Qiu, Eds., Springer, Singapore, 2021.

[37] T. A. Assegie and P. S. Nair, "Handwritten digits recognition with decision tree classification: a machine learning approach," *International Journal of Electrical Computer Engineering*, vol. 9, no. 5, p. 4446, 2019.

[38] K. Mohiuddin and J. Mao, "A comparative study of different classifiers for handprinted character recognition," *Machine Intelligence and Pattern Recognition*, vol. 16, pp. 437–448, 1994.

[39] M. Masood, T. Nazir, M. Nawaz, A. Javed, M. Iqbal, and A. Mehmood, "Brain tumor localization and segmentation using mask RCNN," *Frontiers of Computer Science*, vol. 15, no. 6, article 156338, 2021.

[40] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1–9, Boston, MA, USA, 2015.

[41] S. Ali, J. Li, Y. Pei, M. S. Aslam, Z. Shaukat, and M. Azeem, "An effective and improved CNN-ELM classifier for handwritten digits recognition and classification," *Symmetry*, vol. 12, no. 10, p. 1742, 2020.

[42] S. Ali, Z. Shaukat, M. Azeem, Z. Sakhawat, T. Mahmood, and K. ur Rehman, "An efficient and improved scheme for handwritten digit recognition based on convolutional neural network," *SN Applied Sciences*, vol. 1, no. 9, p. 1125, 2019.

[43] I. J. Goodfellow, Y. Bulatov, J. Ibarz, S. Arnoud, and V. Shet, "Multi-digit number recognition from street view imagery using deep convolutional neural networks," 2013, http://arxiv.org/abs/1312.6082.

[44] M. Uzair and N. Jamil, "Effects of hidden layers on the efficiency of neural networks," in *2020 IEEE 23rd International Multitopic Conference (INMIC)*, pp. 1–6, Bahawalpur, Pakistan, 2020.

[45] K. Urazoe, N. Kuroki, T. Hirose, and M. Numa, "Combination of convolutional neural network architecture and its learning method for rotation-invariant handwritten digit recognition," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 16, no. 1, pp. 161–163, 2021.

[46] N. S. Rani, N. Chandan, A. S. Jain, and H. Kiran, "Deformed character recognition using convolutional neural networks," *International Journal of Engineering Technology*, vol. 7, no. 3, pp. 1599–1604, 2018.

[47] S. Arora and M. S. Bhatia, "Handwriting recognition using deep learning in Keras," in *2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN)*, pp. 142–145, Greater Noida, India, 2018.

[48] R. F. Alvear-Sandoval and A. R. Figueiras-Vidal, "On building ensembles of stacked denoising auto-encoding classifiers and their further improvement," *Information Fusion*, vol. 39, pp. 41–52, 2018.

[49] M. Shatnawi and S. Abdallah, "Improving handwritten arabic character recognition by modeling human handwriting distortions," *ACM Transactions on Asian Low-Resource Language Information Processing*, vol. 15, no. 1, pp. 1–12, 2016.

[50] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10781–10790, Seattle, WA, USA, 2020.

[51] M. Govindarajan, "Evaluation of ensemble classifiers for handwriting recognition," *International Journal of Modern Education Computer Science*, vol. 5, no. 11, pp. 11–20, 2013.

[52] R. M. Cruz, G. D. Cavalcanti, and T. I. Ren, "Handwritten digit recognition using multiple feature extraction techniques and classifier ensemble," in *17th International Conference on Systems, Signals and Image Processing*, pp. 215–218, 2010.

[53] K. Z. Dine, M. Moussaoui, S. Benchaou, and F. Aouinti, "Digit recognition using different features extraction methods," in *Europe and MENA Cooperation Advances in Information and Communication Technologies*, pp. 167–175, Springer, 2017.

[54] P. Phokharatkul and C. Kimpan, "Recognition of handprinted Thai characters using the cavity features of character based on neural network," in *IEEE. APCCAS 1998. 1998 IEEE Asia-Pacific Conference on Circuits and Systems. Microelectronics and Integrating Systems. Proceedings (Cat. No. 98EX242)*, pp. 149–152, Chiang Mai, Thailand, 1998.

[55] A. Singh, M. K. Dutta, M. ParthaSarathi, V. Uher, and R. Burget, "Image processing based automatic diagnosis of glaucoma using wavelet features of segmented optic disc from fundus image," *Computer Methods and Programs in Biomedicine*, vol. 124, pp. 108–120, 2016.

[56] Y. Hou and H. Zhao, "Handwritten digit recognition based on depth neural network," in *2017 International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)*, pp. 35–38, Okinawa, Japan, 2017.

[57] V. Pham, T. Bluche, C. Kermorvant, and J. Louradour, "Dropout improves recurrent neural networks for handwriting recognition," in *2014 14th International Conference on Frontiers in Handwriting Recognition*, pp. 285–290, Hersonissos, Greece, 2014.

[58] B. Shi, X. Bai, and C. Yao, "An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 11, pp. 2298–2304, 2017.

[59] Y. Wang, R. Wang, D. Li, D. Adu-Gyamfi, K. Tian, and Y. Zhu, "Improved handwritten digit recognition using quantum *K*-nearest neighbor algorithm," *International Journal of Theoretical Physics*, vol. 58, no. 7, pp. 2331–2340, 2019.

[60] N. A. Arbain, M. S. Azmi, A. K. Muda, N. A. Muda, and A. R. Radzid, "Offline handwritten digit recognition using triangle geometry properties," *International Journal of Computer Information Systems Industrial Management Applications*, vol. 10, pp. 87–97, 2018.

[61] M. S. Azmi, K. Omar, M. F. Nasrudin, B. Idrus, and K. Wan Mohd Ghazali, "Digit recognition for Arabic/Jawi and Roman using features from triangle geometry," *AIP Conference Proceedings*, vol. 1522, no. 1, pp. 526–537, 2013.

[62] S. Aly and S. Almotairi, "Deep convolutional self-organizing map network for robust handwritten digit recognition," *IEEE Access*, vol. 8, pp. 107035–107045, 2020.

[63] C. J. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[64] A. Jain, G. R. K. Sai Subrahmanyam, and D. Mishra, "Rotation invariant digit recognition using convolutional neural network," *Proceedings of 2nd International Conference on Computer Vision & Image Processing. Advances in Intelligent Systems and Computing, vol 703*, B. Chaudhuri, M. Kankanhalli, and B. Raman, Eds., Springer, Singapore, 2018.

[65] H. Malik and N. Roy, "Extreme learning machine-based image classification model using handwritten digit database," in *Applications of Artificial Intelligence Techniques in Engineering. Advances in Intelligent Systems and Computing, vol 697*, H. Malik, S. Srivastava, Y. Sood, and A. Ahmad, Eds., Springer, Singapore, 2019.

[66] R. Huang, J. Pedoeem, and C. Chen, "YOLO-LITE: a real-time object detection algorithm optimized for non-GPU computers," in *2018 IEEE International Conference on Big Data (Big Data)*, pp. 2503–2510, Seattle, WA, USA, 2018.

[67] M. Nawaz, T. Nazir, M. Masood et al., "Analysis of brain MRI images using improved cornernet approach," *Diagnostics*, vol. 11, no. 10, p. 1856, 2021.

[68] T. Nazir, M. Nawaz, J. Rashid et al., "Detection of diabetic eye disease from retinal images using a deep learning based CenterNet model," *Sensors*, vol. 21, no. 16, p. 5283, 2021.

[69] S. Albahli, T. Nazir, A. Irtaza, and A. Javed, "Recognition and detection of diabetic retinopathy using DenseNet-65 based faster-RCNN," *Computers, Materials and Continua*, vol. 67, no. 2, pp. 1333–1351, 2021.

[70] M. Nawaz, M. Masood, A. Javed et al., "Melanoma localization and classification through faster region-based convolutional neural network and SVM," *Multimedia Tools and Applications*, vol. 80, no. 19, pp. 28953–28974, 2021.

[71] M. Nawaz, T. Nazir, A. Javed et al., "An efficient deep learning approach to automatic glaucoma detection using optic disc and optic cup localization," *Sensors*, vol. 22, no. 2, p. 434, 2022.

[72] S. Pashine, R. Dixit, and R. Kushwah, "Handwritten digit recognition using machine and deep learning algorithms," 2021, http://arxiv.org/abs/2106.12614.

[73] V. Athila and A. S. Chandran, "Comparative analysis of algorithms used in handwritten digit recognition," *International Research Journal of Engineering and Technology*, vol. 8, no. 6, 2021.

[74] H. H. Zhao and H. Liu, "Multiple classifiers fusion and CNN feature extraction for handwritten digits recognition," *Granular Computing*, vol. 5, no. 3, pp. 411–418, 2020.

[75] E. A. Enriquez, N. Gordillo, L. M. Bergasa, E. Romera, and C. G. Huélamo, "Convolutional neural network vs traditional methods for offline recognition of handwritten digits," in *Advances in Physical Agents. WAF 2018. Advances in Intelligent Systems and Computing, vol 855*, R. Fuentetaja Pizán, Á. García Olaya, M. Sesmero Lorente, J. Iglesias Martínez, and A. Ledezma Espino, Eds., Springer, Cham, 2019.

[76] D. Y. Ge, X. F. Yao, W. J. Xiang, X. J. Wen, and E. C. Liu, "Design of high accuracy detector for MNIST handwritten digit recognition based on convolutional neural network," in *2019 12th International Conference on Intelligent Computation Technology and Automation (ICICTA)*, pp. 658–662, Xiangtan, China, 2019.

[77] A. Beikmohammadi and N. Zahabi, "A hierarchical method for Kannada-MNIST classification based on convolutional neural networks," in *2021 26th International Computer Conference, Computer Society of Iran (CSICC)*, pp. 1–6, Tehran, Iran, 2021.

[78] A. K. Agrawal, "Design of CNN based model for handwritten digit recognition using different optimizer techniques," *Turkish Journal of Computer Mathematics Education*, vol. 12, no. 12, pp. 3812–3819, 2021.