

Deep Learning Based Residual Network Features for Telugu Printed Character Recognition

Vijaya Krishna Sonthi^{1,*}, S. Nagarajan¹ and N. Krishnaraj²

¹Department of Computer Science and Engineering, FEAT, Annamalai University, Chidambaram, 608002, Tamilnadu, India

²Department of Networking and Communications, School of Computing, SRM Institute of Science & Technology, Kattankulathur, 603203, Tamilnadu, India

*Corresponding Author: Vijaya Krishna Sonthi. Email: vijayakrishna1990@gmail.com

Received: 07 January 2022; Accepted: 10 February 2022

Abstract: In India, Telugu is one of the official languages and it is a native language in the Andhra Pradesh and Telangana states. Although research on Telugu optical character recognition (OCR) began in the early 1970s, it is still necessary to develop effective printed character recognition for the Telugu language. OCR is a technique that aids machines in identifying text. The main intention in the classifier design of the OCR systems is supervised learning where the training process takes place on the labeled dataset with numerous characters. The existing OCR makes use of patterns and correlations to differentiate words from other components. The development of deep learning (DL) techniques is useful for effective printed character recognition. In this context, this paper introduces a novel DL based residual network model for printed Telugu character recognition (DLRN-TCR). The presented model involves four processes such as preprocessing, feature extraction, classification, and parameter tuning. Primarily, the images of various sizes are normalized to 64x64 by the use of the bilinear interpolation technique and scaled to the 0, 1 range. Next, residual network-152 (ResNet 152) model-based feature extraction and then Gaussian native Bayes (GNB) based classification process is performed. The performance of the proposed model has been validated against a benchmark Telugu dataset. The experimental outcome stated the superiority of the proposed model over the state of art methods with a superior accuracy of 98.12%.

Keywords: Optical character recognition; telugu; deep learning; printed character recognition; residual network

1 Introduction

Optical character recognition (OCR) is defined as the digital conversion of handwritten text into a system-readable form. OCR is a well-known and significant field of pattern analysis model that has been applied in various real-time applications. Initially, OCR was proposed by Carley from the invention of a retina scanner and referred to as an image transmission module [1]. However, in the case of handwritten documents, the writing style may differ from one another, and so the process of character recognition



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

becomes complicated. Hence, handwritten script analysis is assumed to be significant research in recent times. The task of character recognition depends upon the tools and techniques, which evaluates the prediction accuracy of the CR approach. Additionally, OCR is defined as a computation-based field that has implied better enhancements in past decades. Such improvements are accomplished with dense improvements in machine learning (ML) and processing-intensive methods.

The CR system is operated with the aid of resembling human readable action by retaining the accuracy at maximum speed. This model is computed in various phases. Initially, it addresses the recognition process and data acquisition, where the handwritten documents are converted into digitalized form. then, the converted digitalized documents undergo preprocessing under the application of frequent operations on the input image and are converted into effectual form which is applicable for further computation. Preprocessing is mainly performed to remove noise and alternate outliers. In the case of segmentation, text has been classified as lines, and then lines are divided as words. Therefore, the segmentation method is classified into 2 types according to the segment of words such as Holistic and analytical approaches.

The initial category is composed of segmentation-free model, and the second category contains a segmentation reliant mechanism. In the case of the holistic model, words remain characters, and the recognition model finds the string as a complete word, whereas in the case of the analytical approach, words are segmented as individual characters. Thus, the analytical approach is further classified as direct segmentation and indirect segmentation. Initially, a word undergoes segmentation using numerous heuristics and is changed into a single character. Second, strings are segmented into primitives by classifying massive portions with limited character-like subcharacters and small strokes. Once segmentation is completed, feature extraction is carried out, which is the exigent and promising issue in pattern recognition crises. It is subjected to extracting pertinent and exclusive patterns, which limits data for examination and increases the recognition speed. Then, the success rate of a classification method relies on the extracted features.

Different types of miscellaneous languages have been applied globally. Moreover, numerous languages have been lost and their application is reduced, and presence of geographically inaccessible portions are present. Here, OCR and natural language processing (NLP) models are employed for the purpose of resolving language destruction issues. OCR systems are prominently used for languages with better significance and economic values, such as Chinese and latin. Most of the languages are derived from Indic scripts, which are prone to be lost due to insufficient work. Hence, there is an eminent requirement for CR-based developments for Indic scripts.

This paper introduces a novel deep learning (DL)-based residual network model for printed Telugu character recognition (DLRN-TCR). The presented model encompasses three processes: preprocessing, feature extraction, classification, and parameter tuning. First, the images of various sizes are normalized to 64x64 using the bilinear interpolation technique and ranged from (0, 1). Subsequently, residual network-152 (ResNet 152) model-based feature extraction and Gaussian naive Bayes (GNB)-based classification processes are performed. An extensive experimental results analysis takes place on benchmark Telugu dataset. The rest of the paper is defined in the following. Section 2 elaborates the reviewed works related to the study. Section 3 briefly presents the DLRN-TCR model, and Section 4 validates the performance of the presented model. At last, Section 5 finalizes the study.

2 Related Works

In Manisha et al. [2], developers presented the glyph segmentation model for offline handwritten Telugu characters. The key objective of this work is to examine offline hand written Telugu characters with the help of OCR [3]. It is a well-known and crucial mechanism with numerous challenging issues of pattern analysis. It is concentrated in the OCR model for Telugu documents with 3 phases called preprocessing, feature

extraction, and classification. In the first stage, median filtering (MF) is applied to input characters and subjected to normalization as well as skeletonization approaches to extract the edge pixel points. In the second phase, a character is segmented, and the respective centroids for zones are derived. Then, the characters can be identified in various styles. Next, the horizontal and vertical symmetric representation is showcased from a closer pixel of character that has been considered to be binary external symmetry axis constellation for unlimited handwritten characters. Finally, support vector machine (SVM) and quadratic discriminate classifier (QDA) are employed as classification models.

In Prakash et al. [4], the promising issues of OCR in Telugu handwritten text are addressed, and 3 major contributions are developed: (i) database of Telugu characters, (ii) DL-based OCR model, and (iii) client server solution for online development of a model. From the sake of Telugu people and developers, the newly developed code is available freely in this link. In Fields et al. [5], morphological analysis was employed for analyze handwritten Telugu composite characters. Moreover, the composite character in a paper is scanned with an optimal scanning device and developed as a dataset. At this point, datasets are used in preprocessing models. Only simple features are considered for better examination. Then, a handwritten Telugu composite character undergoes preprocessing and segmentation under the application of morphological processes and gains an optimal recognition rate.

In Muppalaneni [6], a handwritten Telugu Compound Character named Guninthalu (Character developed with integration of Telugu Vowels and Consonants) was proposed for the recognition process. Then, classification becomes a more challenging operation. There are several ML frameworks, but accuracy is one of the major challenges to be achieved. In Burra et al. [7], 2 new models were presented for managing these characteristics. Initially, traditional ML models were applied in which 2 class SVMs were employed in word segmentation of words as valuable glyphs in 2 phases. Then, it depends upon the spatial organization of the predicted components. Moreover, it relies on the intuition of valid characters to represent specific patterns in the spatial arrangement of bounding boxes. When the rules are used for acquiring these arrangements, it is necessary to develop an effective model for symbol segmentation.

In Devarapalli et al. [8], diverse employment of fringe maps is illustrated for line segmentation. Traditionally, fringes are employed to perform classification as well as line segmentation. Initially, it can be dealt with joined characters and exploit segmentation free models that do not require prior segmentation of word images as characters or joined elements. In Angadi et al. [9], researchers recommended and estimated a conventional convolutional neural network (CNN) to find online Telugu characters. The network is composed of 4 layers, namely, rectified linear units (ReLUs), softmax activation functions, max pooling and dense layers.

In Phanriam et al. [10], the significance of online language recognition software is defined, and developers have applied various distinguished journals and magazines to learn the value of newly developed tools. Moreover, the requirement for developing online Telugu Character recognition software is increasingly progressed. Then, with the advent of speech interface software, digital ink reliant writing software has been assumed to be a revolutionary modification in the user and machine interaction. Such materials are assumed to be key data, and inputs are compared with keying information with the help of a key board. The major benefits of using these devices and software are that they are noise free and no ergonomic complexities exist.

3 The Proposed DLRN-TCR Model

Fig. 1 shows the block diagram of the presented DLRN-TCR model. The presented model initially reads the image and preprocesses it to normalize the image size. Second, the ResNet 152 model is applied for the extraction of the useful set of features. Finally, the GNB classifier is applied to determine the class of the printer Telugu character. The newly developed method is composed of 2 parts: training and recognition.

Initially, the training portion contributes to data preprocessing, constructing a network structure and network training with preprocessed data. It also contributes to examining the character with the help of the trained method. The detailed processes involved in the presented DLRN-TCR model are discussed below.

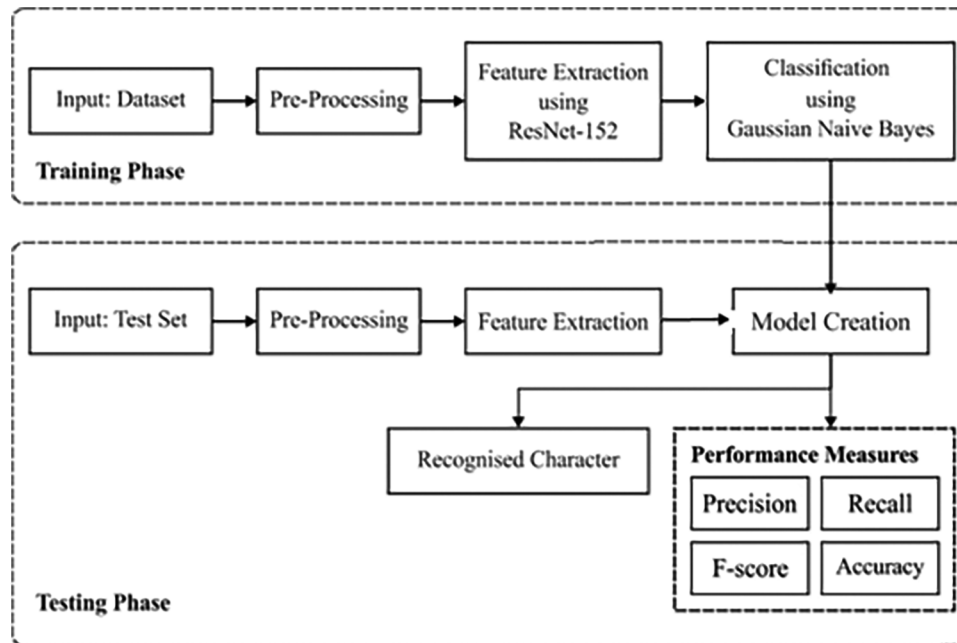


Figure 1: Working process of DLRN-TCR model

3.1 Preprocessing

The images having different sizes are normalized under the application of the bilinear interpolation technique and range from 0, to 1. Here, training is performed in 2 sets of inputs; the first set has actual images, and the second set has inverted images. Therefore, no major difference was attained by means of accuracy or training time.

3.2 CNN Model

In general, CNNs are extensively applied DL approaches in managing image based processes such as image recognition, classification, and captioning. The structure of CNN is depicted in Fig. 2. There are 3 blocks applied in developing the CNN model by different counts of blocks, either by adding or removing the block.

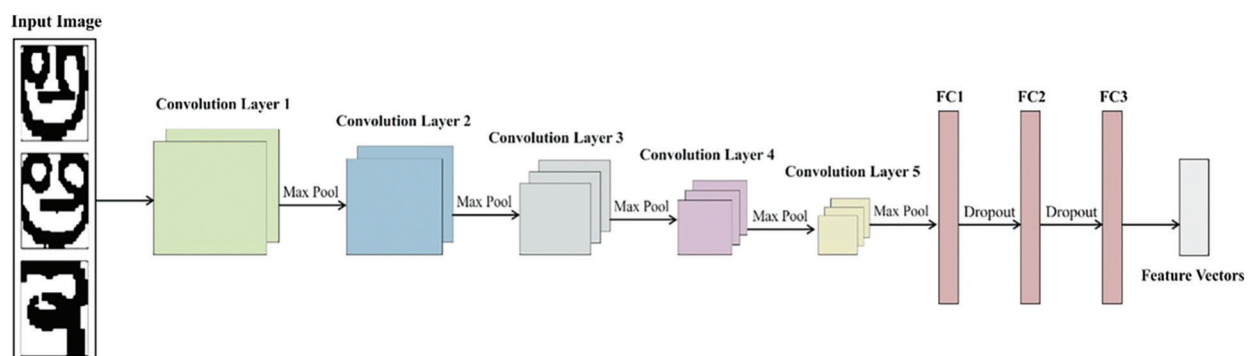


Figure 2: Structure of CNN

A. Convolution layer

This layer varies from neural network (NN) in which limited pixels are connected to the subsequent layer with weight and bias, in that the entire image is classified into small regions while weights and bias are attributed. Moreover, the filters are considered simple ‘features’ that are identified from the input image [11]. The number of filters, size of the local region, strides, and padding are assumed to be the hyperparameters in this layer.

B. Pooling layer

To reduce the spatial dimension of an image and the number of parameters, a pooling layer was applied. This layer computes a fixed function from the input with no parameters. Different types of pooling layers are average pooling, stochastic pooling, and max pooling.

C. Fully connected layer

It acts like a CNN in which the neuron of the existing layer is associated with the current layer. Therefore, the number of parameters is maximum when compared with the convolution layer. This layer is connected to the final layer named the classifier.

D. Activation function

Nonlinear activation functions such as ReLU, leaky ReLU (LReLU), parametric ReLU (PReLU), and Swish have approved the outcomes compared with traditional sigmoid functions. Nonlinear functions are suitable for boosting the training process. Here, developers have applied various activation functions and identified ReLU to be highly efficient when compared with other models.

3.3 ResNet-50 Model

ResNet applies the residual block for resolving the degradation as well as gradient diminishing issues from traditional CNNs. Here, the residual block sharpens the network depth and enhances the function of the network. It is apparent that ResNet systems have accomplished better results in the ImageNet [12] classification process. Moreover, the residual block in ResNet executes the residual method by adding the input of the residual block as well as the result of the residual block. Hence, the residual function is expressed as:

$$y = F(x, W) + x \tag{1}$$

where x implies the input of the residual block; W denotes the weight; and y refers to the result of the residual block. The architecture of a block is represented in Fig. 3.

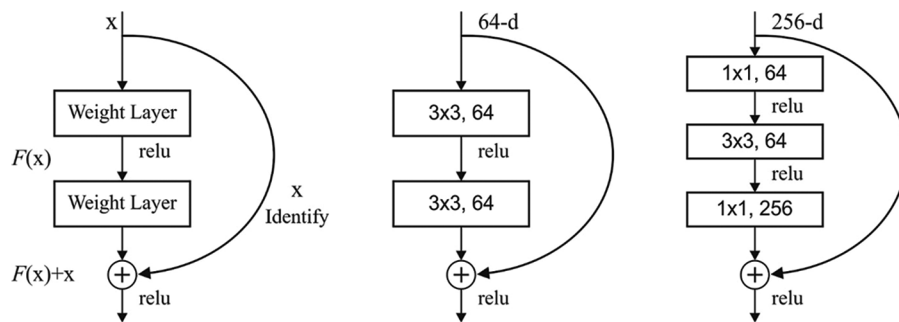


Figure 3: Layers in ResNet: (a) Residual block, (b) 2 layer deep, (c) 3-layer deep

Basically, the ResNet system is composed of various residual blocks where the convolution kernel size differs from the convolution layer. The conventional infrastructures of ResNet are ResNet18, ResNet50, and ResNet101. The fundamental architecture of ResNet50 is applied as depicted in Fig. 4. Thus, the softmax function is generally applied for classification.

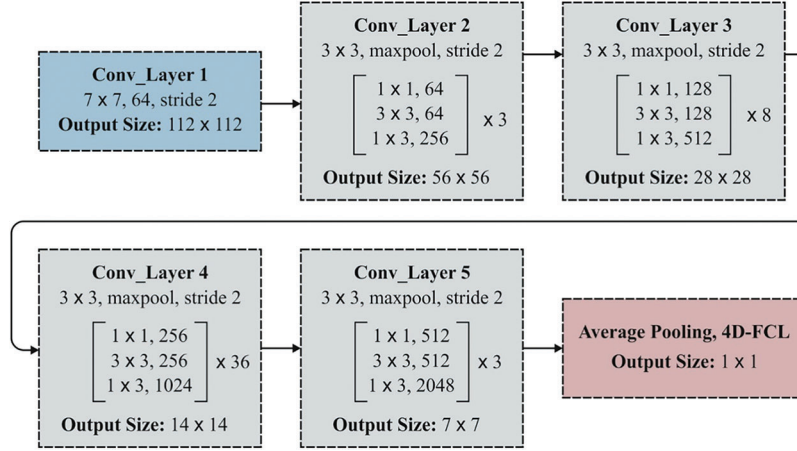


Figure 4: Fundamental layered framework of ResNet-50 model

3.4 Hyperparameters

Multiple hyperparameters are essential for network implementation, few models are structure based, and some modules require efficient network training. Diverse hyperparameters such as stride, padding, and depth are applied for all layers, which are tuned to develop an optimal framework. Moreover, a stride value of 1 was applied to slide one pixel across the input image from the convolutional and pooling layers. Moreover, zero padding is utilized to retain the actual input shape from the output where no data have been dropped at the edges. Finally, Xavier stimulation is applied for weights.

3.5 GNB Model

In general, naive Bayes (NB) is considered a group of supervised learning models that apply Bayes' theorem and the "naïve" assumption of independence [13]. The NB classifier estimates the possibility for a given sample that comes under a specific class. From the instance X , defined by feature vector (x_1, x_n) , and class target y , Bayes' theorem enables us to represent conditional probability $P(y|X)$ as the combination of simple probabilities:

$$P(y|X) = \frac{P(y)P(X|y)}{P(X)} \quad (2)$$

$$= \frac{P(y) \prod_{i=1}^n P(x_i|y)}{P(X)}$$

As $P(X)$ is meant to be the constant for the applied sample, the following rule has been applied for sample classification:

$$\hat{y} = \arg \max_y P(y) \prod_{i=1}^n P(x_i|y) \quad (3)$$

Maximum a posteriori (MAP) evaluation is applied for parameter estimation and attributes in NB approach, along with $P(y)$ and $P(x_i|y)$. In addition, GNB executes the classifier by considering the likelihood of Gaussian features:

$$P(x_i|y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{(x_i - \mu_y)^2}{2\pi\sigma_y^2}\right), \quad (4)$$

where the parameters σ_y and μ_y are evaluated by progressive likelihood. Because of its simplicity and robustness, it is compared with many other effective models, where GNB is used for TCR.

4 Performance Validation

The proposed method is validated utilizing the UHTelPCC dataset, which has 70 K instances of 325 classes, and the samples are classified into 50 K, 10 K, 10 K training, validation, and test sets. UHTelPCC is made from Telugu printed fiction books are scanned at 300 dpi. These books are reproduced in various fonts. Fig. 5 illustrates the sample Telugu document page image [14].

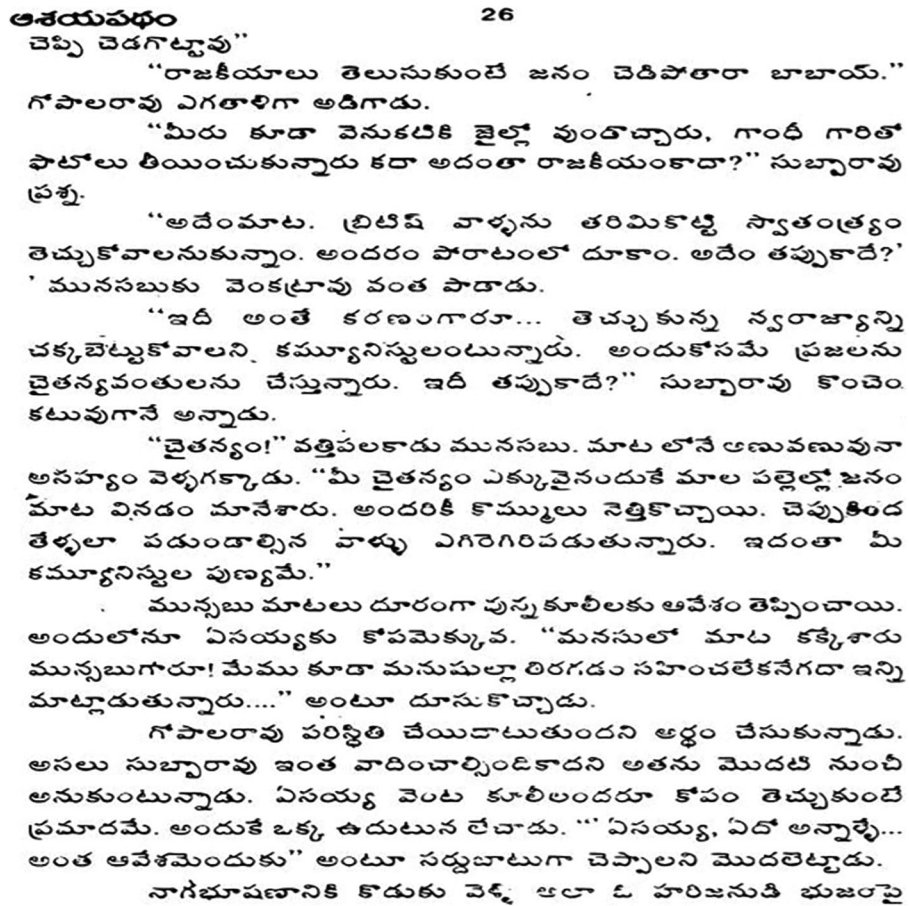


Figure 5: Sample book page

Fig. 6 illustrates the training and validation accuracy of the presented DLRN-TCR model on the applied dataset under varying epoch count. The figure stated that the training accuracy is seemed to be high and the

validation accuracy exceeds the training accuracy on the higher number of epochs. The presented DLRN-TCR model reached a maximum validation accuracy of 0.985.

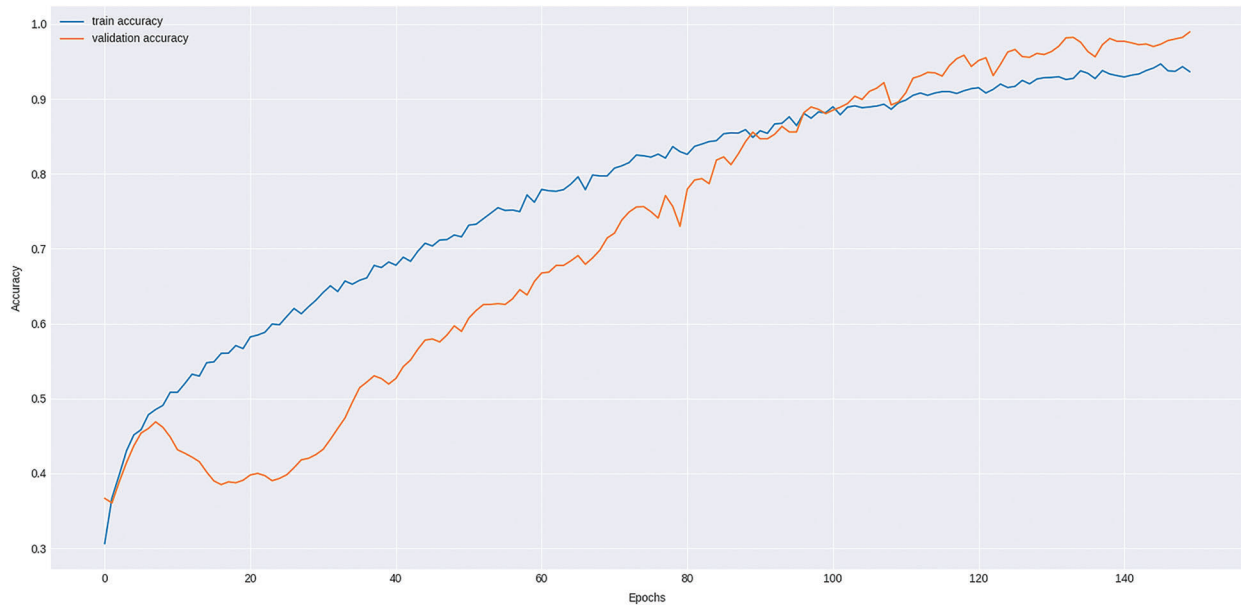


Figure 6: Training accuracy and validation accuracy of the proposed DLRN-TCR method

Fig. 7 shows the training and validation loss of the proposed DLRN-TCR method on the applied dataset under varying epoch counts. The figure shows that the training loss seems to be minimal and that the validation loss exceeds the training loss on a superior number of epochs. The proposed DLRN-TCR technique attained a lower validation loss of 0.085.

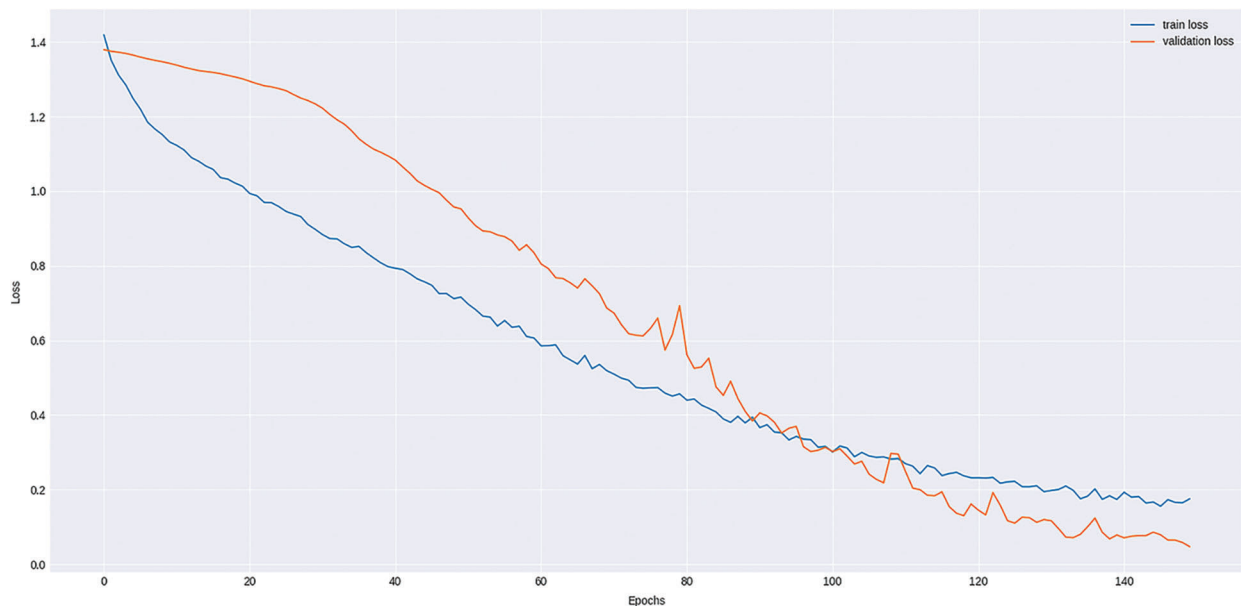


Figure 7: Training loss and validation loss on proposed DLRN-TCR method

Tab. 1 and Figs. 8–10 examines the TCR results analysis of the DLRN-TCR model interms of precision, recall, F-score, and accuracy [15–17]. The overall results analysis signified that the DLRN-TCR model s resulted in a maximum precision of 97.56%, recall of 98.80%, F-score of 97.85%, and accuracy of 98.12%.

Table 1: Result analysis of existing with proposed DLRN-TCR method

Methods	Precision	Recall	F-score	Accuracy
DLRN-TCR	97.56	98.80	97.85	98.12
CNN	95.67	95.82	95.72	96.32
KNN	94.15	95.43	95.12	95.85
MLP	95.72	95.90	95.78	96.45
NN	96.34	96.85	96.80	97.50
DNN	90.89	91.44	91.32	92.10

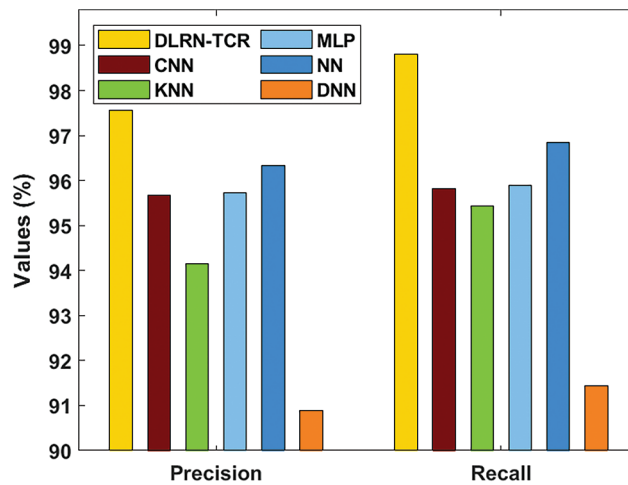


Figure 8: Result analysis of DLRN-TCR model interms of precision and recall

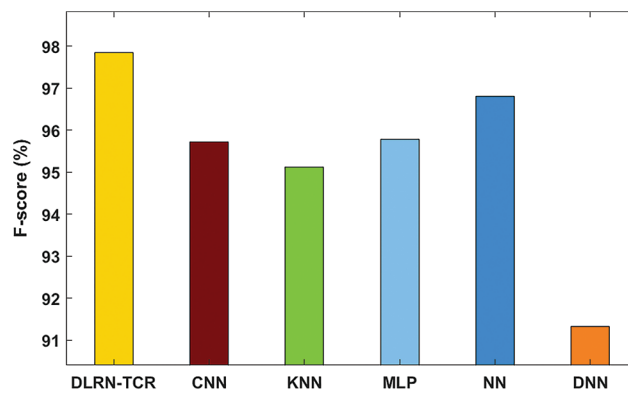


Figure 9: Result analysis of DLRN-TCR model interms of F-score

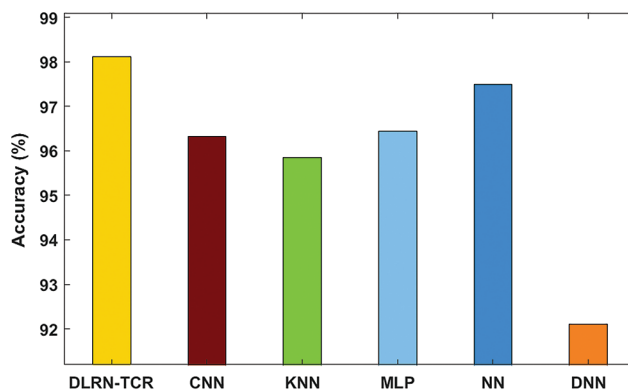


Figure 10: Result analysis of DLRN-TCR model interms of accuracy

On examining the character recognition results interms of precision, it is evident that the DNN model has accomplished inferior outcomes with the least precision of 90.89%. At the same time, the K-Nearest Neighbor (KNN) model resulted in a slightly improved precision of 94.15%. Likewise, the CNN and MLP models led to moderately closer precisions of 95.67% and 95.72% respectively. Similarly, the NN model exhibited competitive performance with a precision of 96.34%. However the presented DLRN-TCR model outperformed the previous methods by attaining a maximum precision of 97.56%. On investigative the character recognition outcomes with respect to recall, it can be evident that the DNN method has accomplished inferior outcomes with a minimum recall of 91.44%. Likewise, the KNN method resulted in a somewhat increased recall of 95.43%. In line with this, the CNN and Multi Layer Perceptron (MLP) methodologies led to moderately closer recalls of 95.82% and 95.90% respectively. Additionally, the NN approach demonstrated competitive performance with a recall of 96.85%. However, the proposed DLRN-TCR method out-performed the previous techniques by obtaining a maximum recall of 98.80%.

On determining the character recognition results interms of F-score, it is evident that the DNN manner has accomplished inferior result with the worst F-score of 91.32%. Simultaneously, the KNN model resulted in a slightly enhanced F-score of 95.12%. Along with that, the CNN and MLP algorithms led to moderately closer F-scores of 95.72% and 95.78% respectively. In addition the NN model has depicted competitive performance with an F-score of 96.8%. Then, the presented DLRN-TCR technique demonstrated the previous models by achieving the highest F-score of 97.85%.

Upon examining the character recognition results with respect to accuracy, it is evident that the DNN model accomplished an inferior effect with the lowest accuracy of 92.10%. Concurrently, the KNN method resulted in a slightly developed accuracy of 95.85%. Moreover, the CNN and MLP methods led to moderately closer accuracies of 96.32% and 96.45% respectively. Furthermore, the NN algorithm exhibited competitive performance with an accuracy of 97.50%. Eventually, the proposed DLRN-TCR model showcased the previous techniques by reaching a superior accuracy of 98.12%. Therefore, it can be employed as an effective tool for TCR. The proposed DLRN-TCR model has achieved effective outcomes due to the inclusion of data normalization, ResNet-152 based feature extraction, and GNB based classification.

5 Conclusion

This paper has developed a new DLRN-TCR model for the recognition of printed Telugu characters. The presented model encompasses four processes: preprocessing, feature extraction, classification, and parameter tuning. The presented model first reads the image and then preprocesses it to normalize the image size. Next,

the ResNet 152 model is employed to extract the collection of feature vectors which are then fed into the GNB classifier to determine the class of the printer Telugu character. The performance of the proposed model has been validated against a benchmark Telugu dataset. The experimental outcome stated the superiority of the proposed model over the state of art methods interms of accuracy, precision, recall, and F-score. As a part of future research, advanced deep learning architectures can be used for printed Telugu character recognition.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

- [1] J. Mantas, "An overview of character recognition methodologies," *Pattern Recognition*, vol. 19, no. 6, pp. 425–430, 1986.
- [2] C. N. Manisha, Y. S. Krishna and E. S. Reddy, "Glyph segmentation for offline handwritten telugu characters," in *Data Engineering and Intelligent Computing*, Singapore: Springer, pp. 227–235, 2018.
- [3] N. Prameela, P. Anjusha and R. Karthik, "Off-line telugu handwritten characters recognition using optical character recognition," in *Proc. Int. Conf. of Electronics, Communication and Aerospace Technology(ICECA)*, Coimbatore, India, vol. 2, pp. 223–226, 2017.
- [4] K. C. Prakash, Y. M. Srikar, G. Trishal, S. Mandai and S. S. Channappayya, "Optical character recognition (OCR) for telugu: Database, algorithm and application," in *Proc.25th IEEE Int. Conf. on Image Processing (ICIP)*, Athens, Greece, pp. 3963–3967, 2018.
- [5] G. Fields and G. D. Vaddeswaram, "Handwritten telugu composite character recognition using morphological analysis," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 18, pp. 667–676, 2018.
- [6] N. B. Muppalaneni, "Handwritten telugu compound character prediction using convolutional neural network," in *Proc. Int. Conf. on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Vellore, India, pp. 1–4, 2020.
- [7] S. Burra, A. Patel, C. Bhagvati and A. Negi, "Improved symbol segmentation for telugu optical character recognition," in *Int. Conf. on Intelligent Systems Design and Applications*, WA, USA, pp. 496–507, 2017.
- [8] K. R. Devarapalli and A. Negi, "Telugu word segmentation using fringe maps," in *Proc. Workshop on Document Analysis and Recognition*, Singapore, pp. 87–96, 2018.
- [9] A. Angadi, V. K. Vatsavayi and S. K. Gorripati, "A deep learning approach to recognize handwritten telugu character using convolution neural networks," *International Journal of Information Systems & Management Science*, vol. 1, no. 2, pp. 1–18, 2018.
- [10] V. K. Sonthi, S. Nagarajan and N. Krishnaraj, "A review on existing learning techniques applied in solving optical character recognition problem," *International Journal of Innovative Computing and Applications*, vol. 12, no. 4, pp. 183–188, 2021.
- [11] B. R. Kavitha and C. Srimathi, "Benchmarking on offline handwritten tamil character recognition using convolutional neural networks," *Journal of King Saud University-Computer and Information Sciences*, vol. 2019, pp. 1–21, 2019.
- [12] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, NV, USA, pp. 770–778, 2016.
- [13] W. Lou, X. Wang, F. Chen, Y. Chen, B. Jiang *et al.*, "Sequence based prediction of DNA-binding proteins based on hybrid feature selection using random forest and Gaussian naive Bayes," *PloS one*, vol. 9, no. 1, pp. 86703–86812, 2014.
- [14] R. Kummariand and C. Bhagvati, "UHTelPCC: A dataset for telugu printed character recognition," in *Int. Conf. on Recent Trends in Image Processing and Pattern Recognition*, Singapore, pp. 24–36, 2018.

- [15] V. K. Sonthi, S. Nagarajan and N. Krishnaraj, “Automated telugu printed and handwritten character recognition in single image using aquila optimizer based deep learning model,” *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 12, pp. 597–604, 2021.
- [16] C. V. Lakshmi, R. Jain and C. Patvardhan, “OCR of printed telugu text with high recognition accuracies,” in *Computer Vision, Graphics and Image Processing*, 1st ed., Berlin, Heidelberg: Springer, pp. 786–795, 2006.
- [17] A. K. Pujari, C. D. Naidu, M. S. Rao and B. C. Jinaga, “An intelligent character recognizer for telugu scripts using multiresolution analysis and associative memory,” *Image and Vision Computing*, vol. 22, no. 14, pp. 1221–1227, 2004.