Advances in Artificial Intelligence and Machine Learning; Research 4 (3) 2517-2534 Received 27-04-2024; Accepted 24-08-2024; Published 1-09-2024

Intelligent Character Recognition Framework for Kannada Scripts via Long Short Term Memory with Thresholding-based Segmentation

Supreetha Patel Tiptur Parashivamurthy supreetha.patel@kittiptur.ac.in
Department of Electronics and Communication Engineering
Kalpataru Institute of Technology, Tiptur 572201
India

Dr. Sannangi Viswaradhya Rajashekararadhya svraradhya@gmail.com
Department of Electronics and Communication Engineering
Kalpataru Institute of Technology, Tiptur 572201
India

Corresponding Author: Supreetha Patel Tiptur Parashivamurthy

Abstract

Various opinions were made by the researchers to develop an automatic network for Optical Character Recognition (OCR). Still, character recognition in handwritten scripts is an unsolved task. In this paper, two efficient techniques are developed an effective character recognition technique for the handwritten Kannada scripts. The Kannada Character Recognition (KCR) techniques faced several challenges due to the different writing styles of people and the absence of fixed spacing among alphabets, words and lines. Another complication in the KCR model is the absence of large datasets to train the network, and it isn't easy to write the Kannada script by combining the Kannada alphabets. Therefore, a new handwritten KCR approach is developed to identify the characters from the ancient Kannada scripts. The required Kannada script images are gathered from various online databases. The garnered images are preprocessed and segmented using morphological operation and thresholding. The relevant features from the images are achieved by the geometric feature extraction method. Finally, the characters are recognized by utilizing the Long Short Term Memory (LSTM) network, and the experimental results will be analyzed over the traditional optimization strategies and baseline works to evaluate the efficiency of the proposed network.

Keywords: Character recognition, Kannada script, Long short term memory, Morphological operation and thresholding.

1. INTRODUCTION

Character recognition is a procedure that enables computers to recognize the characters from written or printed scripts, where the characters in the numbers or letter format are converted into an understandable format for the computer. The primary goal of character recognition is to modify a text image into a text format that can be read by machines [1]. Character recognition is most

Citation: Supreetha Patel Tiptur Parashivamurthy and Dr. Sannangi Viswaradhya Rajashekararadhya. Intelligent Character Recognition Framework for Kannada Scripts via Long Short Term Memory with Thresholding-based Segmentation. Advances in Artificial Intelligence and Machine Learning. 2024;4(3):147.

commonly used for scanning printed materials into editable versions that may be modified with word processors such as Microsoft Word or Google Docs, automatic data extraction, processing, and entry and indexing of print material for search engines [2]. Kannada character set consists of huge classes due to the presence of vowel modifiers and consonants [3]. The basic form of Kannada script is completely different from Roman script. Kannada script is more difficult than English script due to the existence of compound characters. Still, this script does not use the upper-case/lower-case concept. The primary goal is to recognize online handwritten scripts that contain words, characters, writings, lines, paragraphs, and so on. There is broad work in the field of handwritten script recognition, along with various existing analyses [4].

The electronic or mechanical exchange of handwritten or printed characters from photos or documents, scanned documents, subtitles on an image or from a scene photo into machine-readable text is known as OCR [5]. OCRs produce better performance in one script's contents that results in worse performance for other scripts. The effectiveness of the OCR relies upon the document layout, script type, writing styles and printed/handwritten datasets [6]. Segmentation issues vary for every language. Moreover, the issues related to handwritten characters also vary from user to user. Hence, it is necessary to improve the OCR framework to obtain accurate and effective results during segmentation [7]. The grouping of characters from the scripts is performed based on deep learning and machine learning techniques. The machine learning techniques attained better performance in classifying the Kannada character classes; however, they provide accurate results for only a few specific classes. Therefore, research efforts to classify a set of characters are performed by utilizing deep learning techniques [8].

Even though machine learning models show promising results for several complex recognition problems, such as character recognition, deep learning techniques are more efficient in handling many functions and multiclass problems. KCR using deep learning techniques provides better outcomes than traditional machine learning techniques. Deep learning techniques provide advanced results in segmentation and also while processing large datasets when compared with other techniques [9]. Application of deep learning techniques on complex computer vision and pattern recognition techniques provides successful results. The neural network plays a vital role in handwritten KCR. Recently, neural networks have been utilized in various types of pattern recognition [10]. KCR performance can be improved by utilizing deep learning techniques such as neural networks. Thus, the KCR framework is developed for Kannada scripts by utilizing deep learning techniques like LSTM along with thresholding-based segmentation.

The major contributions of the developed KCR model are given below.

- To implement an effective KCR system for Kannada scripts using advanced deep learning technique to recognize the character more accurately.
- To utilize deep learning such as LSTM for recognizing the Kannada characters with a higher accuracy rate, which also supports performing with less computational complexity.
- To investigate the effectiveness of the introduced framework by examining it with different recognition techniques.

The implemented KCR system is described below. The merits and demerits of the traditional models are explained in second section. An elaborate explanation of the implemented framework is shown in the third section. The fourth section deals with the character recognition framework based on LSTM and threshold-based segmentation. The fifth section deals with the discussion and the result obtained for the proposed model, and the sixth section provides the conclusion of the implemented framework.

2. LITERATURE SURVEY

2.1 Related Works

In 2021, Sridevi and Rangarajan [11], have implemented a Deep Convolutional Neural Network (DCNN) for recognizing the characters from a printed Kannada script. Several characters in the scripts were affected by some degradation that led to breakage and dilations of the printed character, which introduced various challenges in the recognition process. The developed model contains three stages. The result produced by this level was taken as the input for the fully connected layer where the classification of characters takes place. This developed model has utilized the character with 156 classes and each class with 100 instances. Sixty iterations with 4 epochs were used to calculate the effectiveness. At the training phase, a high classification rate was noted.

In 2022, Rani *et al.* [12], have introduced a deep network with capsule and routing layers to achieve a better accuracy rate in the recognition of Kannada characters from a handwritten script. A Capsule Network (CN) was developed with an input layer, primary capsule, trilevel dense convolution layer, routing capsule layers, two convolution layers and an output layer. This model utilized the datasets that were gathered from more than 100 users, which were used to create 7769 training data samples containing 49 classes. Test samples were collected from 3 to 5 users from each 49 classes to generate 245 samples for novel patterns. On the basis of performance analysis, the classification process resulted in a loss, but high precision with better accuracy was achieved for 43 classes. The remaining 6 classes achieve an average precision with average accuracy.

In 2023, Supreetha and Sannangi [13], have developed an ensemble-based framework for the KCR. Necessary handwritten Kannada text images were initially gathered. After pre-processing these collected images with filtering methods and CLAHE, the active contour technique was used to segment the collected images. The segmented images were used with code-based histogram methods and the adaptive local tetra pattern. “Hybrid Honey Badger Henry Gas Solubility Optimization (HGSO)” was used to optimize the Local Tetra Pattern (LTrP) parameters. Finally, the Ensemble learning model was used to recognize handwritten letters from the Kannada scripts. In addition, the HGSO + HBA algorithm was utilized to determine the rate of recognition along with the parameter tuned in the deep learning approach. The designed method obtained better accuracy during the analysis. As a result, the proposed method established improved performance on various performance metrics.

In 2010, Pal and Singh [14], have implemented a “Multi-Layer Perceptron Network (MLPN)” with a hidden layer for an English Handwritten Character. Boundary tracing and Fourier Descriptor were the features that were extracted from the handwritten character. A given character can be determined by comparing its properties and by analyzing the shape of each character.. 500 samples of handwriting from both male and female participants of varying ages were used to train the system. The test was carried out on 500 samples other than the samples used to train the system that has denoted as the combination of a back propagation network and a Fourier description produced better recognition accuracy for handwritten English characters in a short training period.

In 2018, Rani *et al.* [15], have investigated on the recognition of Kannada scripts. The proposed experiment was based on the recognition of handwritten character images collected from a variety of unrestricted environments, and degraded character images were collected from old Kannada documents. It led to several inconsistent behaviors of the recognition algorithm that had reduced the recognition accuracy. AlexNet was utilized as the deep convolution neural network to train the degraded patterns from character image samples. The handwritten datasets were collected from users aged between 18 to 21, 22 to 25, and 26-30, and the printed data collected from ancient Kannada poetry/literature images were utilized in the performance evaluation of this proposed model. There were nearly 497 classes in the datasets. Still, semantic analysis was performed at the post-processing stage of character recognition for the compound characters that do not overlap or touch. The Alex net utilized in the recognition of printed characters produced better results with high accuracy.

2.2 Problem Statement

The main challenges in character recognition from a handwritten script are the quality and dissimilarity of the text images taken from the script. The quality of the character image may vary based on the time. Different people have different handwriting. But, sometimes, the same people may have different handwriting. In such cases, it isn't easy to recognize the handwritten character accurately. Several networks were developed to recognize the character from a Kannada script. The advantages and disadvantages of a few traditional KCR techniques are illustrated in TABLE 1. DCNN [11], is capable of handling large datasets and providing more accurate results. Still, it requires a large amount of computational resources and is expensive. CN [12], is capable of protecting the hierarchies among the features and requires less data for training. Yet, it doesn't classify simple and multi-compound characters, and it is not tested with a large data set such as ImageNet. HGSO [13], provides better performance in order to avoid local optimum. However, it has reduced exploitation ability. MLPN [14], is capable of solving complex nonlinear problems and producing fast prediction results after training. Yet, this method is difficult to implement and requires more parameters for processing. CNN [15], provides highly scalable results and is capable of handling large datasets. Still, it has a high computation cost, and it requires more training data. Therefore, it is essential to develop an effective framework to resolve issues faced in the existing techniques and also to improve character recognition in Kannada scripts using deep learning approaches.

3. INTELLIGENT CHARACTER RECOGNITION FRAMEWORK FOR KANNADA SCRIPTS VIA DEEP LEARNING

3.1 Proposed Kannada Script-Based Character Recognition Framework

Character recognition from a handwritten Kannada script is a function that identifies the character from an image source. Kannada character from a handwritten script is difficult to identify due to the different handwriting of different peoples. The space between the characters and the writing styles affects the recognition techniques to achieve better performance in recognition. The main challenge in utilizing machine learning techniques in character recognition of Kannada scripts is it

Table 1: Features and challenges of Existing KCR techniques

Author [citation]	Methodology	Features	Challenges
Sridevi and Rangarajan [11]	DCNN	• This method is capable of handling large datasets. • This method provides more accurate results.	• It requires a large amount of computational resources. • It is expensive.
Rani *et al*. [12]	CN	• This method is capable of protecting the hierarchies among the features. • This method requires less data for training.	• It doesn't classify simple and multi-compound characters. • It is not tested with a large data set such as ImageNet.
Supreetha and Sannangi [13]	HGSO	• This method provides better performance in order to avoid local optimum.	• It has reduced the exploitation ability.
Pal and Singh [14]	MLPN	• This method is capable of solving complex nonlinear problems. • This method produces fast prediction results after training.	• This method is difficult to implement. • It requires more parameters for processing.
Rani *et al*. [15]	CNN	• This method provides highly scalable results. • This method is capable of handling large datasets.	• It has a high computation cost. • It requires more training data.

requires large data sets to train the system. "Support Vector Machine (SVM)" is the most commonly used KCR technique in character recognition. But, it cannot provide accurate results due to the robust features of the network. Due to these challenges, deep learning techniques are introduced for performing character recognition. In OCR, deep learning techniques are commonly utilized. It scans the character from a script and transforms it into machine language. This technique provides better performance and improves the accuracy of the recognized characters. Still, if the quality of the document is low, it is difficult for the OCR software to recognize and extract the text accurately. Therefore, an enhanced deep learning technique is introduced to resolve these challenges in the existing character recognition system. The architectural representation of the developed character recognition model is depicted in FIGURE 1.

The implemented KCR framework for Kannada scripts is to recognize the Kannada characters more effectively. Kannada characters are difficult to recognize due to their varying handwriting. Hence, a recognition network is implemented by utilizing a deep learning mechanism along with the thresholding for segmentation. Initially, the framework collects the raw Kannada script images from an online dataset. The gathered images are given to the pre-processing phase. The images are preprocessed using the morphological operations and given into the segmentation phase, where segmentation is performed using the thresholding technique. Then, the features are extracted by utilizing the "Gray Level Co-Occurrence Matrix (GLCM)" and the region properties of the images. The extracted features are given to the LSTM network for recognizing the Kannada characters.

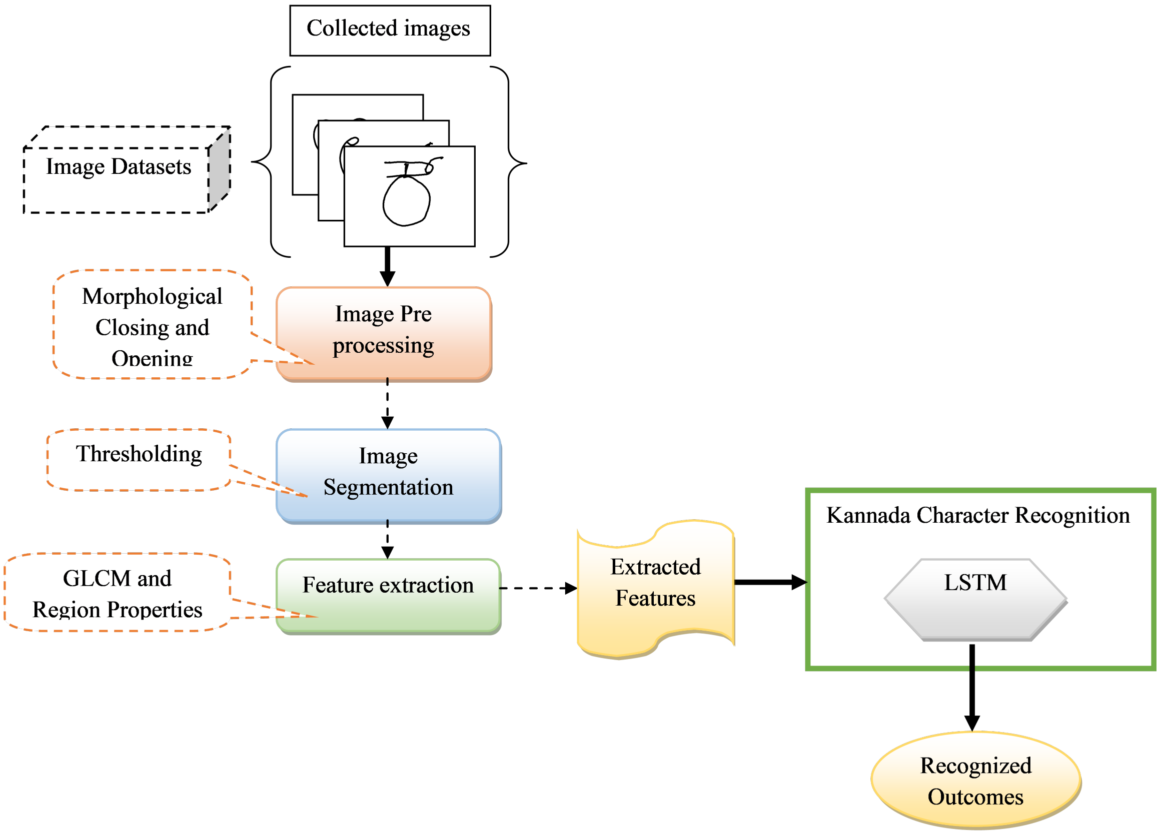

Figure 1: Architectural representation of developed Kannada character recognition system using Kannada scripts

3.2 Kannada Handwritten Character Image Collection

The implemented KCR framework utilizes the handwritten character images from a "Kannada Handwritten Characters Dataset". The dataset link is given below: "https://www.kaggle.com/datasets/dhruvildave/kannada-characters?resource=download&select=Img". This dataset consists of around "16,425 handwritten images" of the Kannada language. It is a classification dataset, which is commonly used for the classification task performed using computers. It contains 657 groups, and every group holds 25 images. The images are stored in .png format. The gathered images are denoted as . The sample images from the dataset are given in FIGURE 2.

3.3 Image Pre-processing

In the image "pre-processing stage", collected images img_c^{inp} are taken as input. Image pre-processing is the technique of formatting an image before the images are subjected to inference and training. Image preprocessing is utilized to improve the quality of the image in order to increase the effective analysis of the image. In this developed model, the character images are pre-processed using the "Morphological opening and closing technique". "Morphological opening and closing" is the

Image description	Kannada characters	Recognized characters
1	ಕ	a
2	ಖ	kha
3	ನ್ನ	nna
4	ಕೊ	o
5	ಕು	u

Figure 2: Sample images of Kannada characters from the dataset

technique used to improve the image quality. This technique is the result of a fundamental operation of dilation and erosion. In the opening phase, the collected images are eroded and then dilated. At this phase, a tiny substance from the front is removed and placed in the background of the image. In the closing phase, the images are dilated and then eroded. At this phase, tiny holes in the front transfer the tiny particles from the background to the foreground of the image. The pre-processed images are obtained as the output, and it is denoted by img_c^{pre}.

4. Long Short Term Memory for Character Recognition Framework in Kannada Scripts With Thresholding-Based Segmentation

4.1 Thresholding-Based Segmentation

In the “segmentation stage”, the pre-processed images img_c^{pre} are taken as input. Image thresholding plays an essential part in pattern recognition and image processing. Thresholding [16], is commonly used in the image segmentation process. They provide segmented results by easy experimentation, and the results are a sequence of continuous segments. This technique determines a threshold value T_s that builds a border between the gray level range of the image parallel to the object and a range equal to the background of the image. Later, the thresholding of the gray-level image is altered to the binary. There are a few algorithms that utilize more than one threshold value and allocate pixel values to one class rather than two classes.

The process of threshold segmentation is given below. Initially, the threshold value T_s of every pixel on the image is determined. If the “gray value” is higher than the threshold value T_s, then it is fixed at the target point with the value 1, or else, it is fixed at the background point with the value 0. In the case of programming, the value at the target point is taken as 255 and the value at the background point is taken as 0. Thus, the image is divided into two sections, namely, “target region” and “background region”. The arithmetic form of threshold-based segmentation is given in

Eq. (1).

$$g(a,b) = \begin{cases} 1 & if\ f(a,b) > T_s \\ 0 & if\ f(a,b) \leq T_s \end{cases} \tag{1}$$

From Eq. (1), the phrase T_s refers threshold value of the image, and the terms (a, b) represent the pixels of an image. After the segmentation process, the segmented image is gained as the output, and it is denoted by img_c^{seg}.

4.2 Feature Extraction

The segmented images img_c^{seg} are considered as the input in the "feature extraction stage". Feature Extraction is referred to as the transformation of original information into a numerical feature. Further, it can be utilized for processing only if the details stored in the original dataset are preserved. In this implemented network, the features are extracted by utilizing GLCM and region properties.

GLCM [17], is referred to as a technique utilized to examine the texture of an image. It analyzes the spatial relationship among the pixels and determines how the coordination of pixels is found in the image. GLCM is described as a square matrix with order O. The GLCM-based textural features include correlation, homogeneity, energy and contrast.

Correlation: It is defined as the measures of joint probability that occur on a specified pixel pair. The arithmetic representation of correlation is given in Eq. (2).

$$Correlation = \frac{\sum_{l=0}^{pq-1} \sum_{m=0}^{pq-1} -n_{l,m} * \log\, n_{l,m}}{\sigma_a \sigma_b} \tag{2}$$

Homogeneity: It is the "measurement of contact of the distributed components in the GLCM to the GLCM diagonal. The arithmetical illustration of the homogeneity is specified in Eq. (3).

$$Homogeneity = \sum_{l,m=0}^{O-1} \frac{n_{l,m}}{1 + (l - m)^2} \tag{3}$$

Energy: It is referred to as the "sum of squared elements in the GLCM". The statistical demonstration of the Energy is shown in Eq. (4).

$$Energy = \sum_{l,m=0}^{O-1} \left(n_{l,m}\right)^2 \tag{4}$$

Contrast: It is referred to as the measurement of local variation on the GLCM. The numerical representation of the contrast is shown in Eq. (5)

$$Contrast = \sum_{l,m=0}^{O-1} m_{l,m}\, (l - m)^2 \tag{5}$$

From the above Equations, the term $n_{l,m}$ denotes the elements of the symmetrical GLCM, the term pq represents the level count, and the term σ indicates the variances of the intensity.

Region Properties [18]: The region is referred to as the closed area in an image. The region includes several properties like centroid, bounding box area etc. Region properties are the mathematical features of the specific region of an image. It measures the properties of a labeled region. The "region props" produce the value of the measurement in the form of 'pixels'. A few of the region properties are Area, Perimeter, Extrema, Filled Area, MinorAxisLength, Convex Image, Extent, Convex Area, Eccentricity, Centroid, Filled Image, Pixel List, MajorAxisLength, SubarrayIdx, PixelIdxList, Solidity, Convex Hull, bounding box and EquivDiameter. The extracted features from GLCM and region properties are concatenated and denoted by the term Fe_h^{exf}.

4.3 LSTM-aided Character Recognition

In the LSTM network, the extracted features Fe_h^{exf} are taken as input. LSTM [19], is capable of learning the long dependencies by utilizing the feedback loops. Conventional Recurrent Neural Networks (RNN) investigates overcoming the feed-forward NNs, which is also known as "memory loss" that is responsible for producing less performance on time-series datasets and arrays. This system utilizes periodicity in its "hidden layer" to obtain "Short Term Memory (STM)" and also to obtain the data from the time-series datasets and arrays. Still, RNN is affected by the vanishing gradient problem, which limits the system in learning long-range dependencies. LSTM addresses these issues by storing the procedural information on the memory cells and also hiding the unnecessary data, then obtaining enhanced performance than the conventional RNN.

LSTM network is developed by embedding three gates, namely input gate IP, output gate OP, reset or forget gate FR and candidate cell GP. The LSTM network can create a controlled flow of information by specifying which data should be remembered and forgotten. Later, the "input gate" IP_T along with the "candidate gate" GP_T maintains the data, which is extracted recently and stored in the memory position of the gate GP_T at duration T. The reset or forget gate FR manages the prior data that must either vanish or to save at the memory cell at the interval $T-1$. The output gate OP maintains the data that the memory cells can utilize for obtaining the output.

$$Wl = \begin{bmatrix} Wl_{IP} \\ Wl_{FR} \\ Wl_{GP} \\ Wl_{OP} \end{bmatrix}, \; Rl = \begin{bmatrix} Rl_{IP} \\ Rl_{FR} \\ Rl_{GP} \\ Rl_{OP} \end{bmatrix}, \; Bl = \begin{bmatrix} Bl_{IP} \\ Bl_{FR} \\ Bl_{GP} \\ Bl_{OP} \end{bmatrix} \tag{6}$$

From Eq. (6), the term Wl denotes the weight, the term Rl indicates the recurrent weight, the term Bl represents the biases of the network, and the terms IP, FR, OP, GP indicate the input gate, reset or forget gate, output gate and the candidate cell, respectively. The functions implemented by an LSTM network are shown below. The function performed in the input gate is given in Eq. (7)

$$IP_T = \sigma\left(Wl_{IP}a_T + Rl_{IP}N_{T-1} + Bl_{IP}\right) \tag{7}$$

The function performed in the reset or forgot gate is given in Eq. (8)

$$FR_T = \sigma\left(Wl_{FR}a_T + Rl_{FR}N_{T-1} + Bl_{FR}\right) \tag{8}$$

The function performed in the output gate is given in Eq. (9)

$$OP_T = \sigma\left(Wl_{OP}a_T + Rl_{OP}N_{T-1} + Bl_{OP}\right) \tag{9}$$

The function performed in the candidate cell is given in Eq. (10)

$$GP_T = \tan N \left(Wl_{GP} a_T + Rl_{GP} N_{T-1} + Bl_{GP} \right) \tag{10}$$

From the above equations, the term a_T denotes the input vector given to the LSTM network; the term N_T indicates the hidden state of the network that contains the output of the memory cell. The output of the memory cell is calculated by the Eq. (11)

$$N_T = OP_T \Theta \tan N \left(M_T \right) \tag{11}$$

From Eq. (11), the term Θ represents the component-wise multiple operator, and the sigmoid function σ is computed as shown in Eq. (12).

$$\sigma(a) = \left(1 + e^{-a}\right)^{-1} \tag{12}$$

The memory state M_T of the LSTM unit is evaluated by the Eq. (13)

$$M_T = GP_T \Theta M_{T-1} + IP_T \Theta GP_T \tag{13}$$

The structural representation of LSTM-aided KCR is shown in FIGURE 3.

5. RESULTS AND DISCUSSIONS

5.1 Experimental Setup

The implemented KCR framework was performed on the Python platform. The effectiveness of the development was compared with traditional methods. The performance of the network was examined with prediction techniques like Gated Recurrent Units (GRU) [20], RNN [21], Fermionic Neural Network (FermiNet) [22], and Jordan recurrent network (JRN) [23].

5.2 Performance Measures

The performance measures considered for the examination are explained below,

(a) Accuracy: It is the "amount of accurate prediction among the total case analyzed". The numerical demonstration of accuracy is expressed in Eq. (14).

$$Accuracy = \frac{sa_T + hj_T}{sa_T + hj_T + ax_F + wn_F} \tag{14}$$

(b) Recall: Recall is defined as the true positive rate or sensitivity. The mathematical representation of recall is shown in Eq. (15).

$$Recall = \frac{sa_T}{sa_T + wn_F} \tag{15}$$

(c) Specificity: It is described as "the possibility of hj_T rate in the implemented network. It is numerically shown in Eq. (16).

$$Specificity = \frac{hj_T}{hj_T + ax_F} \tag{16}$$

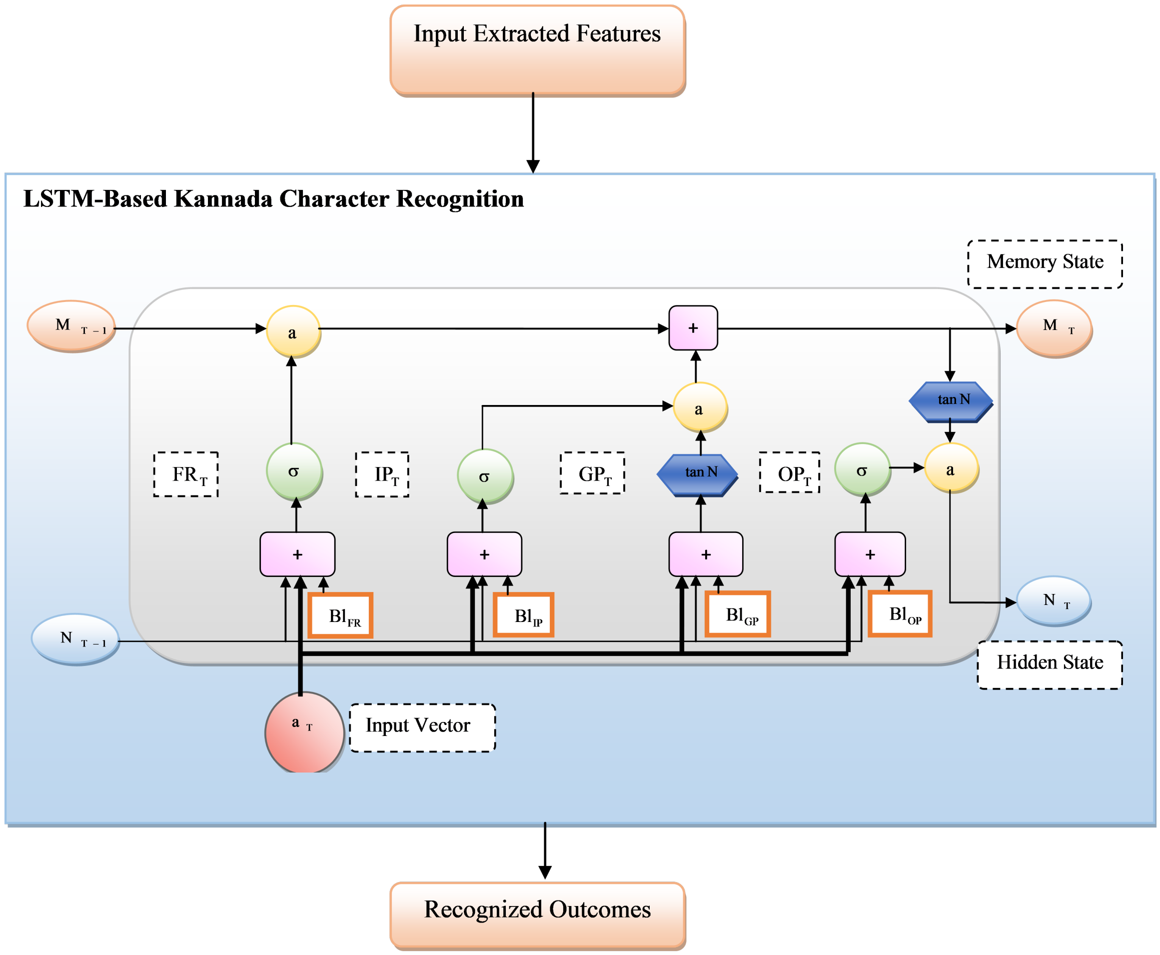

Figure 3: Structural representation of LSTM-aided Kannada Character Recognition

(d) Precision: Precision is referred to as the "accurate positive predictive value". It is arithmetically shown in Eq. (17).

$$\Pr ecision = \frac{sa_T}{sa_T + ax_F} \tag{17}$$

(e)FPR: False Positive Rate is "the probability of the ax_F ratio". The numerical representation of FPR is shown in Eq. (18).

$$FPR = \frac{ax_F}{ax_F + hj_T} \tag{18}$$

(f) FNR: False Negative Rate describes the 'ratio of 'positives' with obtained test outcomes in 'negatives' at the testing phase'. It is numerically shown in Eq. (19).

$$FNR = \frac{wn_F}{wn_F + sa_T} \tag{19}$$

(g) NPV: Negative Predictive Value is defined as the 'flow of cash takes place at various periods. It is shown in Eq. (20).

$$NPV = \frac{hj_T}{ax_F + hj_T} \tag{20}$$

(h) FDR: False Discovery Rate is the "predicted count of false positives among other rates". It is numerically shown in Eq. (21).

$$FDR = \frac{hj_T}{hj_T + sa_T} \tag{21}$$

(i) F1-Score: F1-Score expresses "the middle value lying among precision and recall". The arithmetical representation is given in Eq. (22).

$$F1\ Score = \frac{2 *\ sa_T}{2 * (sa_T + hj_T + wn_F)} \tag{22}$$

(j) MCC: Mathews Correlation Coefficient describes the relation measure on two binary variables. It is calculated in Eq. (23).

$$MCC = \frac{sa_T \times hj_T - ax_F \times wn_F}{\sqrt{(sa_T + ax_F)\,(sa_T + wn_F)\,(hj_T + ax_F)\,(hj_T + wn_F)}} \tag{23}$$

5.3 Recognition Analysis on Developed Framework

The effectiveness analysis of the implemented model by employing different recognition algorithms is shown in FIGURE 4. In Fig 4(d), the FNR of LSTM is 45.45% better than GRU, 53.84% improved than RNN, 14.28% greater than FermiNet, and 33.33% superior to JRN, when the "hidden neuron count" is 75. Hence, performance analysis on the implemented model shows superior performance of the developed model than other recognition techniques.

5.4 ROC Analysis on Developed Model

The ROC analysis for the implemented network is shown in FIGURE 5. In FIGURE 5, the 'true positive rate' of the LSTM is 13.92% advanced than GRU, 12.5% higher than RNN, 9.75% increased than FermiNet, and 5.88% greater than JRN when 'false positive rate' is at 0.6. Thus, the outcome of the ROC analysis of the developed system shows enhanced efficiency than existing techniques.

5.5 Confusion Matrix Analysis on Developed Model

The confusion matrix analysis for the implemented framework is shown in FIGURE 6. The Confusion matrix describes the performance of the recognition technique in a matrix form. The obtained result proved the implemented framework shows improved performance than other techniques.

5.6 Overall Analysis of the Developed Kannada Character Recognition Model

The overall performance examination of the implemented model is given in TABLE 2. In TABLE 2, the MCC of the introduced framework is 46.08% higher than GRU, 62.73% improved than RNN, 13.44% better than FermiNet and 30.04% superior to JRN. Thus, the effectiveness of the introduced network shows an advanced performance compared to other recognition techniques.

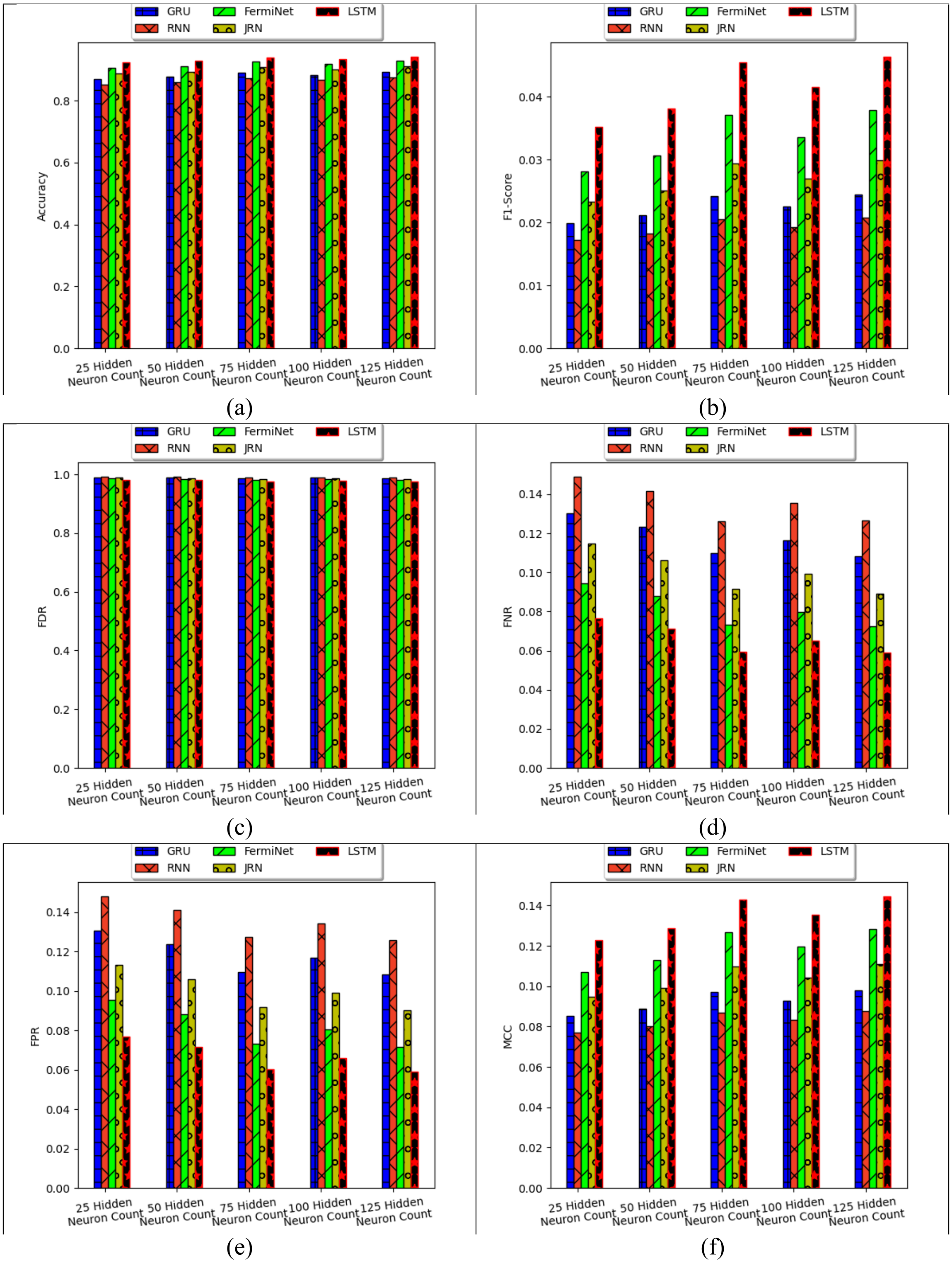

Figure 4: Recognition Analysis on Developed Framework based on (a) Accuracy, (b) F1-score, (c)FDR, (d) FNR, (e) FPR, (f) MCC, (g) NPV, (h) Precision, (i) Recall and (j) Specificity

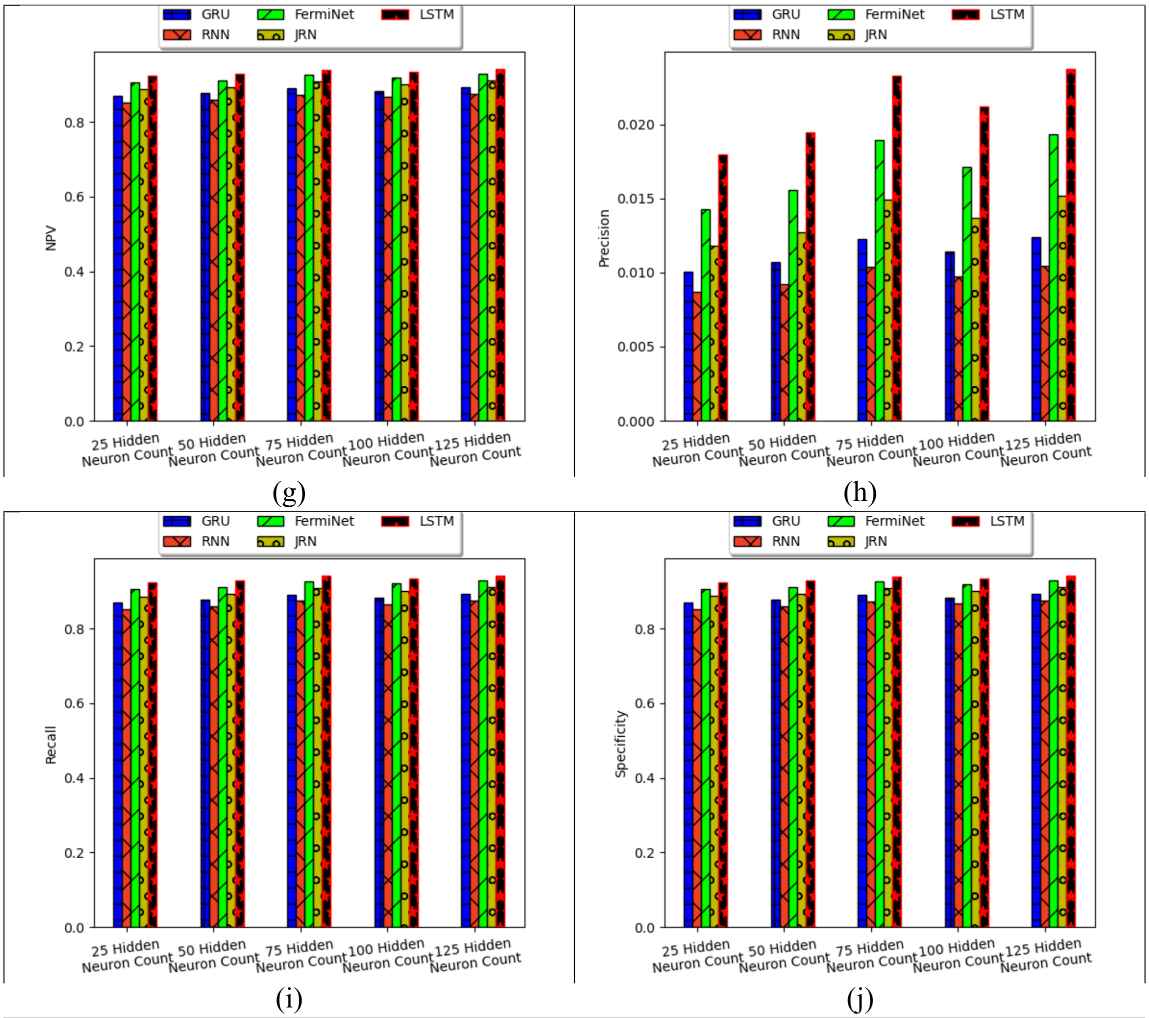

Figure 4: Continued..

The confusion matrix analysis as shown in FIGURE 6 indicates that the implemented LSTM framework outperforms other techniques in classification accuracy and error reduction.

The overall performance evaluation reveals that the MCC of the proposed framework is 46.08% higher than GRU, 62.73% higher than RNN, 13.44% higher than FermiNet, and 30.04% higher than JRN.

The developed LSTM-based framework shows significant improvement in various recognition metrics:

Accuracy: The proposed model's accuracy is 5.76% greater than GRU, 7.89% better than RNN, 1.61% higher than FermiNet, and 3.67% better than JRN.

Recall: The recall rate of the LSTM framework is 13.92% higher than GRU, 12.5% higher than RNN, 9.75% better than FermiNet, and 5.88% greater than JRN at a false positive rate of 0.6.

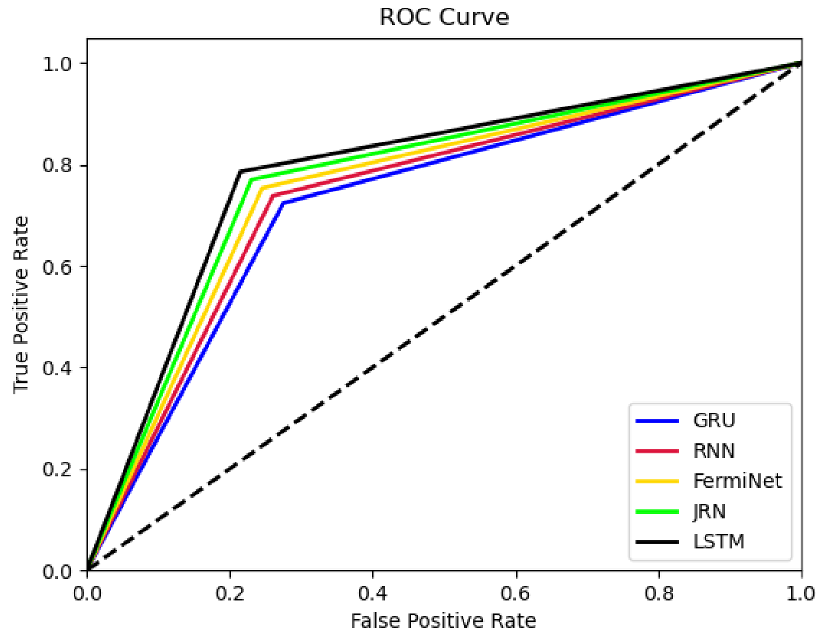

Figure 5: ROC Analysis on Developed KCR Model

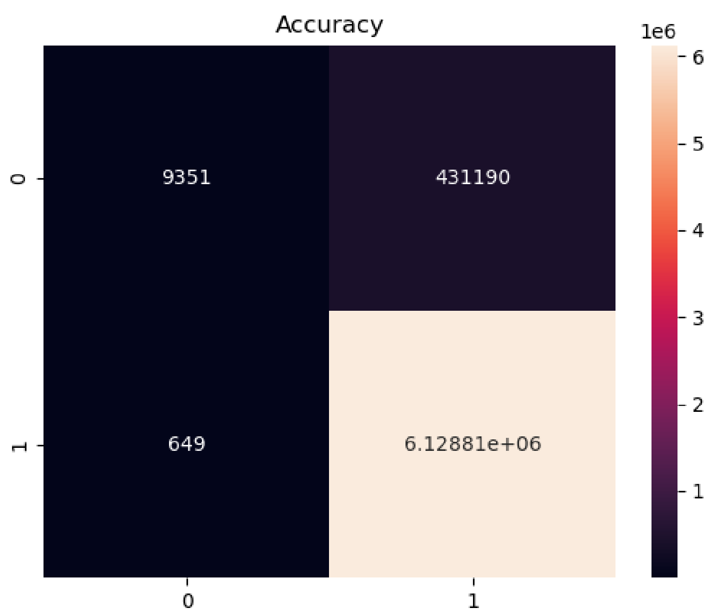

Figure 6: Confusion Matrix of the developed Kannada character recognition model based on the dataset

Table 2: Overall Performance Analysis of the developed model based on different recognition techniques

TERMS	GRU [20]	RNN [21]	FermiNet [22]	JRN [23]	LSTM
FDR	0.9885826	0.9902693	0.9828798	0.9862935	0.9787738
Precision	0.0114174	0.0097307	0.0171202	0.0137065	0.0212262
F1-Score	0.0225435	0.0192447	0.0336152	0.0270022	0.0415101
FNR	0.1163	0.1355	0.0796	0.0992	0.0649
Recall	0.8837	0.8645	0.9204	0.9008	0.9351
NPV	0.8833602	0.8658869	0.9194503	0.9011893	0.9342698
Accuracy	0.8833607	0.8658848	0.9194518	0.9011887	0.9342711
MCC	0.0927573	0.0832645	0.1194471	0.1042014	0.135504
FPR	0.1166398	0.1341131	0.0805497	0.0988107	0.0657302
Specificity	0.8833602	0.8658869	0.9194503	0.9011893	0.9342698

In 2021, Sridevi and Rangarajan [11], implemented a Deep Convolutional Neural Network (DCNN) for printed Kannada script recognition, achieving high classification rates. However, this method was limited to printed scripts and did not address the challenges of handwritten script recognition.

In 2022, Rani et al. [12], developed a Capsule Network (CN) for handwritten Kannada character recognition, achieving high precision and accuracy. Nonetheless, the proposed LSTM-based framework surpasses the Capsule Network in terms of overall performance metrics.

The proposed Kannada Character Recognition (KCR) framework utilizing LSTM and thresholding-based segmentation exhibits superior performance across multiple metrics compared to recent published works. The advanced accuracy, recall, and MCC values confirm the effectiveness of the developed model in recognizing Kannada characters more accurately and efficiently.

6. CONCLUSION

The implemented KCR framework was developed to recognize the Kannada characters from a script more effectively. In the starting phase, the raw images were gathered from online sources. Then, those images were pre-processed by utilizing morphological operation, and segmentation was performed by thresholding. Then segmented images were given to the feature extraction stage, where the features from the images were extracted by the “GLCM and region properties”. Extracted features were transferred to the LSTM network for classifying the Kannada characters from the scripts. The effectiveness of the implemented framework was tested by examining the performance with different recognition techniques. From the evaluation, the accuracy of the introduced was 5.76% larger than GRU, 7.89% improved than RNN, 1.61% enhanced than FermiNet and 3.67% superior to JRN. Thus, the proposed Kannada character recognition framework has given an advanced performance rate when compared with other conventional methods.

References

[1] Sethy A, Patra PK. Deformed Character Recognition Using Convolutional Neural Networks. J Web Eng. 2018;17:3629-3653.

[2] Dhandra BV, Hangarge M, Mukarambi G. Spatial Features for Handwritten Kannada and English Character Recognition. Special Issue on RTIPPR. Int J Comput Appl. 2010;10:146-150.

[3] Prasad MM, Sukumar MS, Ramakrishnan AG. Divide and Conquer Technique in Online Handwritten Kannada Character Recognition. In: Proceedings of the International Workshop on Multilingual OCR. 2009:1-7.

[4] Kunte R, Rao S, Sudhaker Samuel RD. On-Line Character Recognition for Handwritten Kannada Characters Using Wavelet Features and Neural Classifier. IETE J Res. 2000;46:387-393.

[5] Rao AS, Sandhya S, Anusha K, Arpitha CN, Meghana SN. Exploring Deep Learning Techniques for Kannada Handwritten Character Recognition: A Boon for Digitization. Int J Adv Sci Technol. 2020;29:11078-11093.

[6] Fateh A, Fateh M, Abolghasemi V. Multilingual Handwritten Numeral Recognition Using a Robust Deep Network Joint With Transfer Learning. Inf Sci. 2021;581:479-494.

[7] Desai AA. Gujarati Handwritten Numeral Optical Character Reorganization Through Neural Network. Pattern Recognit. 2010;43:2582-2589.

[8] Guptha NS, Balamurugan V, Megharaj G, Sattar KNA, Rose JD. Cross Lingual Handwritten Character Recognition Using Long Short Term Memory Network With Aid of Elephant Herding Optimization Algorithm. Pattern Recognit Lett. July 2022;159:16-22.

[9] Sonu Varghese K, James A, Chandran S Dr. A Novel Tri-Stage Recognition Scheme for Handwritten Malayalam Character Recognition. Procedia Technol. 2016;24:1333-1340.

[10] Sanjeev Kunte R, Sudhaker Samuel RD. A Simple and Efficient Optical Character Recognition System for Basic Symbols in Printed Kannada Text. Sādhanā. 2007;32:521-533.

[11] Sridevi TN, Lalitha Rangarajan. Deep Convolutional Neural Networks for Degraded Printed Kannada Character Recognition. Indian J Comput Sci Eng (IJCSE). 2021;12.

[12] Rani NS, Manohar N, Hariprasad M, Pushpa B. Robust Recognition Technique for Handwritten Kannada Character Recognition Using Capsule Networks. Int J Electr Comp Eng. February 2022;12:383-391.

[13] Parashivamurthy SPT, Rajashekararadhya SV. Recognition of Kannada Character Scripts Using Hybrid Feature Extraction and Ensemble Learning Approaches. Cybern Syst. 2023:1-36.

[14] Pal A, Singh D. Handwritten English Character Recognition Using Neural Network. Int J Comput Sci Commun. 2010;1:141-144.

[15] Shobha Rani NS, Chandan N, Sajan Jain AS, Kiran HR. Deformed Character Recognition Using Convolutional Neural Networks. Int J Eng Technol. 2018;7:1599-1604.

[16] Panwar P, Gopal G, Kumar R. Image Segmentation Using K-Means Clustering and Thresholding. Image. 2016;3:1787-1793.

[17] Öztürk, Ş, Akdemir B. Application of Feature Extraction and Classification Methods for Histopathological Image Using Glcm, Lbp, Lbglcm, Glrlm and Sfta. Procedia Comput Sci. 2018;132:40-46.

[18] Patel S, Trivedi P, Gandhi V, Prajapati GI. 2D Basic Shape Detection Using Region Properties. Int J Eng Res Technol. 2013;2:1147-1153.

[19] Deabes W, Sheta A, Braik M. Ect-Lstm-Rnn: An Electrical Capacitance Tomography Model-Based Long Short-Term Memory Recurrent Neural Networks for Conductive Materials. IEEE Access. 2021;9:76325-76339.

[20] Zia S, Azhar M, Lee B, Tahir A, Ferzund J, et al. Recognition of Printed Urdu Script in Nastaleeq Font by Using Cnn-Bigru-Gru Based Encoder-Decoder Framework. Intell Syst Appl. 2023;18: 200194.

[21] Naz S, Umar AI, Ahmed SB, Ahmad R, Shirazi SH, et al. Statistical Features Extraction for Character Recognition Using Recurrent Neural Network. Pak J Stat. 2018;34.

[22] Zheng PL, Wang JB, Zhang Y. Efficient and Quantum-Adaptive Machine Learning With Fermion Neural Networks. Phys Rev Applied. 2023;20:044002.

[23] Lee SW, Hee-Heon S. A New Recurrent Neural-Network Architecture for Visual Pattern Recognition. IEEE Trans Neural Netw. 1997;8:331-340.