

Malayalam Handwritten Character Recognition using CNN Architecture

Pranav P Nair¹, Ajay James², Philomina Simon³, Bhagyasree.P. V⁴

^{1,2}Department of Computer Science and Engineering, Government Engineering College, Thrissur

³Department of Computer Science and Engineering, University of Kerala, Thiruvananthapuram

⁴Department of Computer Science and Engineering, Christ College of Engineering, Irinjalakuda

Article Info

Article history:

Received May 29, 2023

Revised Sep 3, 2023

Accepted Sep 23, 2023

Keyword:

Character Recognition
Convolutional Neural Network
Malayalam HCR
Neural Network
Pattern Recognition
OCR

ABSTRACT

The process of encoding an input text image into a machine-readable format is called optical character recognition (OCR). The difference in characteristics of each language makes it difficult to develop a universal method that will have high accuracy for all languages. A method that produces good results for one language may not necessarily produce the same results for another language. OCR for printed characters is easier than handwritten characters because of the uniformity that exists in printed characters. While conventional methods find it hard to improve the existing methods, Convolutional Neural Networks (CNN) has shown drastic improvement in classification and recognition of other languages. However, there is no OCR model using CNN for Malayalam characters. Our proposed system uses a new CNN architecture for feature extraction and softmax layer for classification of characters. This eliminates manual designing of features that is used in the conventional methods. P-ARTS Kayyethu dataset is used for training the CNN and an accuracy of 99.75% is obtained for the testing dataset meanwhile a collection of 40 real time input images yielded an accuracy of 95%.

Copyright © 2023 Institute of Advanced Engineering and Science.
All rights reserved.

Corresponding Author:

Ajay James,
Department of Computer Science and Engineering, Government Engineering College,
Ramavarnapuram Engineering College P O,
Thrissur, Kerala 680009
Email: ajayjames80@gmail.com

1. INTRODUCTION

Breakthroughs in pattern recognition challenges were achieved by using deep learning for the recognition of images. Deep learning methods outperform conventional methods in the field of pattern recognition. Deep learning requires a very large data set for training the network. Manual feature designing is complicated, time-consuming, and requires features to be designing by the developer. The effectiveness of the entire system depends on these features. This major issue in conventional methods is solved by deep learning networks by eliminating feature designing and thus the task of feature extraction can be automated. Convolutional neural networks have provided a breakthrough in the IMAGENET challenge 2012. Alex net is the CNN model used and an error rate of 16% was obtained compared to the error rate of 25% in 2011 [1]. Since then, CNN has attracted a lot of attention and has become the state of the art in image recognition tasks. CNN was well suited for image recognition tasks because of its ability to represent the image structure. Local connectivity strategy that makes use of the special structure of the image and weight sharing strategy that reduces the number of parameters involved made CNN more suitable for image recognition.

Recognizing handwritten characters is a difficult task as each character is written in different ways by different people. Apart from the writing styles, other factors like noise and occurrence of skew also add to the challenge. By creating new features, combining characteristics that are previously known, and utilising

various classifiers, researchers have been experimenting with various techniques to improve the accuracy rates of handwritten character recognition. However, these attempts have not resulted in a significant improvement in the recognition rate of characters. Though CNN has produced better accuracy rates, there exist certain challenges that need to be addressed. Lack of proper data set and high hardware requirement is a major hurdle for deep learning.

[Bindu Madhuri](#) et al. [2] proposed a work that depends on a methodical technique to perceive both disconnected and online Telugu manually written characters with leftover learning structure called ResNet. This paper bargains in building up a quick, dependable Telugu transcribed ResNet for both on the web and disconnected character acknowledgment and furthermore improves the order execution. The Proposed lingering net (ResNet-50) accomplishes 2.37% mistake on the ResNet-18 and 34 test set.

[G. Ramesh](#), et al. [3] proposed a work, for recognition of Kannada handwritten words using SVM classifier with convolutional neural network. In this work they use the structural property of characters to obtain a primary segmentation path. Also graph distance theory is used to separate the overlapped or joined characters. Utilising a Support Vector Machine (SVM) classifier, the segmentation results are confirmed in the end. Different databases that comprise both printed and handwritten texts are subjected to thorough simulation. Benchmarking findings show that the proposed algorithms perform better than other traditional methods in terms of accuracy and sensitivity.

Manjusha J et al. [4] proposed a work based on CNN framework for handwritten Malayalam character recognition. Instead of using the time-consuming handcrafted feature extraction methods here automatic feature extraction using CNN is used. This takes less time for training and testing and also gives results which are more error free. The Alex net model of CNN was used in this paper. The work was done for compound Malayalam characters also.

Samatha P Salim et al. [5] proposed a work for recognition of handwritten Malayalam characters using ResNet more efficiently. Despite the fact that it accomplishes around a 0.95 acknowledgment rate, some misclassification exists. This is since the classifier work at the yield layer performs a characterization that doesn't think about the boundary changes utilizing the low and mid-level highlights. This paper proposes an effective strategy for acknowledgment of handwritten Malayalam character including compound characters and signs, utilizing deep layered neural network architecture. This method achieves an accuracy of 99.56% using multi-scaled ResNet architecture for PARTS Kayyethuthu dataset [28].

Bhagyasree P V et al. [6] proposed a framework to recognize handwritten cursive English characters using Directed Acyclic Graph – Convolutional Neural Network [DAG – CNN]. Rather than using the linear CNN architecture, DAG – CNN is used. In basic CNN, output of every convolutional layer is connected to next convolutional layer only. So, only the output from last convolutional layer will be directly connected to softmax layer for classification. But in DAG – CNN architecture, more than one convolutional layer will be directly connected to the softmax layer. Hence it can incorporate more features. This work achieves an accuracy of 97.31% for cursive small letters and an accuracy of 99.56% for cursive capital letters.

Yasin Zamani et al. [7] have used the well-known Random Forest (RF) and convolutional neural network algorithms for investigating Persian handwritten digit recognition on the Hoda dataset, using it as a standard test bed for comparing both the techniques. It was found that CNN can perform better when a large training set is used.

A Deep Learning Based Large Scale Handwritten Devanagari Character Recognition was proposed by Shailesh Acharya et al. [8]. They classified handwritten Devanagari characters using convolutional neural network. Over-fitting in their network was reduced by increasing the size of data set and adding a dropout layer. They built two network configurations and compared their performances. Model A had three convolution and one fully connected layer. However, a shallower network was chosen for model B. Model A achieved the highest testing accuracy of 98.47% when compared to 98.26% of model B.

Jomy John et al. [9] used Wavelet transforms along with Support Vector Machine (SVM) classifier for Malayalam character recognition. The image is converted into a grayscale image and re-scaled so that it fits into 64×64 matrices. 2-D Haar wavelet transform is completed on the re-scaled image and features are extracted. These features were used to train the SVM. Classification accuracy of 89.64% is reported while using third-level decomposition and 90.25% while using second level decomposition. Only 44 characters in Malayalam were considered. Compound characters were not classified in this paper.

Abdul Rahiman M et al. [10] separated the complete character set into three different groups depending on the character's shape and appearance: Ra type characters, Pa type characters, and special symbols. High-Low-High (HLH) values are used to find the length and breadth of each character. An accuracy of 92% was obtained using this method. When noise is present, there is a considerable probability of miss-recognition.

Manju Manuel et al. [11] suggested using a neural network and the curvelet transform to recognise handwritten Malayalam characters. The Malayalam language is a good fit for the curvelet transform because

it offers a concise representation for curved singularities. A Dataset of 53 characters was created by collecting samples from people of different age groups. After computing the projection profile of characters, Curvelet transformations were used to compute the features. Character recognition is achieved using a multi-layer perceptron. An accuracy of 80.79% is reported for feature extraction using Curvelet transform. However, not all characters in the Malayalam character set are considered.

Meiyin Wu et al. [12] explored the two mainstream algorithms of deep learning which are the Convolutional Neural Network and the Deep Belief Network (DBN). A real-time handwritten character dataset and the MNIST database were used to evaluate the performance of CNN and DBN.

Yujie Liu et al. [13] For identifying Chinese and alphanumeric characters in vehicle licence plates, a CNN model with Shared Hidden Layers (SHL) and two different softmax layers has been proposed. To avoid the over-fitting problem in the training process, the early stopping rule is employed. SHL-CNN shows a better performance than conventional CNN. The Absolute Reductions (AR) in error are 0.384% and 0.131% and the Relative Reductions (RR) in error are 9.289% and 9.63% for the Chinese and the alphanumeric characters respectively.

A character recognition system for Tamil characters using CNN was proposed by Prashanth Vijayaraghavan et al. [14]. They used stochastic pooling, local contrast normalization, and probabilistic weighted pooling for learning features by augmenting the ConvNetJS library to achieve an accuracy of 94.4%. IWFHR-10 dataset was used in their work.

Offline Malayalam OCR using multiple Classifiers was proposed by Anitha Mary M.O. Chacko et al. [15]. Gradient-based features and density-based features were used and an accuracy of 81.82% was obtained. Malayalam Character recognition using run-length count and features that were based on gradient was put forward by G. Raju et al. [16]. Another scheme for recognizing characters was also put forward by the author. In this method, they combine local and global features for recognizing isolated Malayalam characters. Gradient features were applied in the recognition process of Malayalam vowels.

Recognition of Devanagari characters using multiple classifiers was proposed by Arora Sandhya et al. [17]. They used moment invariant and chain code histogram in their system. They used projection distance as a feature in addition to zonal based features for getting higher recognition accuracy. The classifier used was the Nearest Neighbour classifier. The entire image was divided into 25 equal zones. The pixel distance from the individual is calculated from the centroid of the image. If multiple pixels are present in a grid column then the average pixel distance is found out and stored. This procedure is repeated for the entire grid to get total of 250 features. This method was reported to give an accuracy of 96.5%.

Another paper proposed by Ajay James et al. [18] suggest a method based on regional zoning and structural features for Malayalam handwritten character recognition. The input to the system is images of Malayalam handwritten characters. The system will recognize the characters and will provide a Unicode to the user. For the feature size of 126, the accuracy obtained is 95.4% and for a feature vector of size 9 is 78.67% of efficiency.

Jino P J et al. [19] proposed a holistic approach for the recognition of offline Malayalam district names. Words are extracted from the input image using the bounding box. A histogram of oriented gradients was selected for feature extraction. The dimensionality of the image is reduced using Principal Component Analysis (PCA), Gaussian Random Projection, and Sparse Random Projection. A dataset was collected from 56 different people. This dataset is divided into training and testing set in the ratio 80:20. The classification was completed using SVM, RF, and Neural networks. Using PCA as the dimensionality reduction technique and SVM as a classifier provided the highest accuracy of 97%.

Ajay James et al. [20] proposed a hybrid technique of feature extraction based on geometrical and structural properties of characters for Malayalam handwritten characters. This method includes two stages. Classification of characters into various groups based on the geometrical features is done in the first stage. Geometrical features include the number of endings, bifurcations and loops and in the second stage the characters are recognized based on specific characteristics defined for each group. The results show that the three-stage recognition scheme based on geometrical and structural properties of character is an efficient method of Malayalam HCR with the recognition rate of 96.50% and accuracy of 95.77% in stage-I and 93.86% in Stage-II.

Jomy John et al. [21] put forward a system for Malayalam handwritten characters. The principal component analysis was used for dimensionality reduction. Gradient and curvature calculation are the major phases in the feature extraction process. Arctangent is found out for the gradient to get directional information and it acts as the gradient feature. Curvature feature was taken as the strength of the gradient in the direction of curvature. SVM was the classifier used for their system. SVM along with the Radial Basis function gave an accuracy of 97.96%.

Meenu Alexa et al.[22] used dissimilar classifiers for Malayalam character recognition. The input image was converted to a grayscale image followed by binarization using Otsu's method. A median filter was

used to remove salt and pepper noise that may be present in the image. Line segmentation was completed using a Horizontal projection profile and the characters are segmented using a vertical projection profile. Speeded-Up-Robust-Features (SURF), curvature, and diagonal features were extracted from the character image. SURF and curvature features were fed into the SVM classifier whereas diagonal features were fed into a neural network. The results of both classifiers were combined to get a better result. A dataset of 33 characters was made for this purpose.

Jomy John et al. [23] Extreme Learning Machine (ELM) and SVM comparisons were also made, along with a brand-new gradient-based feature descriptor. The feature descriptor used in the proposed method is the strength of quantized directional information obtained from a gradient image by using filters. A dataset of handwritten characters in Malayalam which comprises 14,800 images were used in this method. An accuracy of 93.52% was obtained while using the SVM classifier which was higher than using the ELM classifier even though the latter had lesser training time.

Nishad A et al. [24] put forward a method that uses Gabor filters for recognizing offline handwritten Malayalam characters. Gabor based features were extracted using Gabor filters and additional features such as the ratio of horizontal and vertical grid values were considered. An accuracy of 96.80% was obtained for this method that uses a lesser number of uncorrelated images.

Chain code histogram for handwritten character recognition in Malayalam was proposed by Jomy John et al. [25]. Contours are represented as chain codes and Chain code Histogram, which is invariant to scale and translation, was calculated from it and was further normalized. For classification, a 2 layered feed-forward neural network was used and an accuracy of 72.1% was obtained.

A complete OCR system that recognized characters from an input document rather than recognizing just characters was proposed by Shanjana C et al. [26]. The document was binarized using Otsu's technique. A horizontal projection profile was used to segment lines. The average height was used to separate lines that were too close. Word segmentation was done on the assumption that the distance between words is always greater than the distance between characters. Vertical projection profile was used in word segmentation and also in character segmentation with the addition of connected component analysis. SVM was the classifier used along with RBF kernel and for dimensionality reduction, PCA was used. The overall accuracy of 89.7% was reported for this method. There is no work reported on Malayalam handwritten offline character recognition that uses CNN for feature extraction and there was no method that yields 100% accuracy.

Li Chen et al. [27] proposed a framework for recognizing handwritten digits and Chinese characters. There were mainly three parts in the framework namely, the sample generation, CNN models, and voting. The sample generation is realized by local distortion and global distortion. The network structure of the CNN models is designed according to the properties of handwritten characters and several training tricks are also employed for better training. Multiple trained CNN models are used to vote for the final recognition result. Voting can significantly improve the recognition rate. The error rate for the MNIST dataset was about 0.23% with just CNN and 0.2% for humans. In contrast, the proposed framework reported an error rate of 0.18%. The experimental results on CASIA show an error rate of 3.21% while the error rate of the earlier was 5.23%. This method has a very high accuracy rate for recognition of Chinese characters and even achieved a better recognition rate than a human being.

Malayalam is one of the popular languages in south India and is spoken by more than 95% of people in Kerala. In addition to being the official language of Kerala state, it is a member of the Dravidian language family. Malayalam was designated as a Classical Language in India in 2013 and is one of the twenty-two scheduled languages in India. [29]. Characters in the Malayalam language exhibit a curved nature, combinations of certain characters yield new characters and the presence of special characters makes it an arduous task to recognize Malayalam characters. In this paper, CNN is used to achieve a better accuracy rate in recognition of Malayalam handwritten characters.

The rest of the paper is organized as follows. Section 2 is research method; section 3 gives the result and discussion and section 4 has the conclusion.

2. RESEARCH METHOD

In this research work, a Convolutional Neural Network (CNN) for handwritten Malayalam character recognition is built from scratch. All the parameters and layers are chosen to achieve maximum accuracy rates. The kernel weights are initialized using Gaussian distribution. The raw images that are gathered from different sources are scanned and each character is obtained from an individual image. Pre-processing is completed first on the image to make it ready for further steps. The dataset for training and testing is created. Dataset Augmentation is used to create an even more massive dataset than the one which was built. The overall architecture of the proposed system is shown in the following Figure 1.

The proposed system is divided into five modules. The modules are pre-processing, CNN modelling, classification and testing.

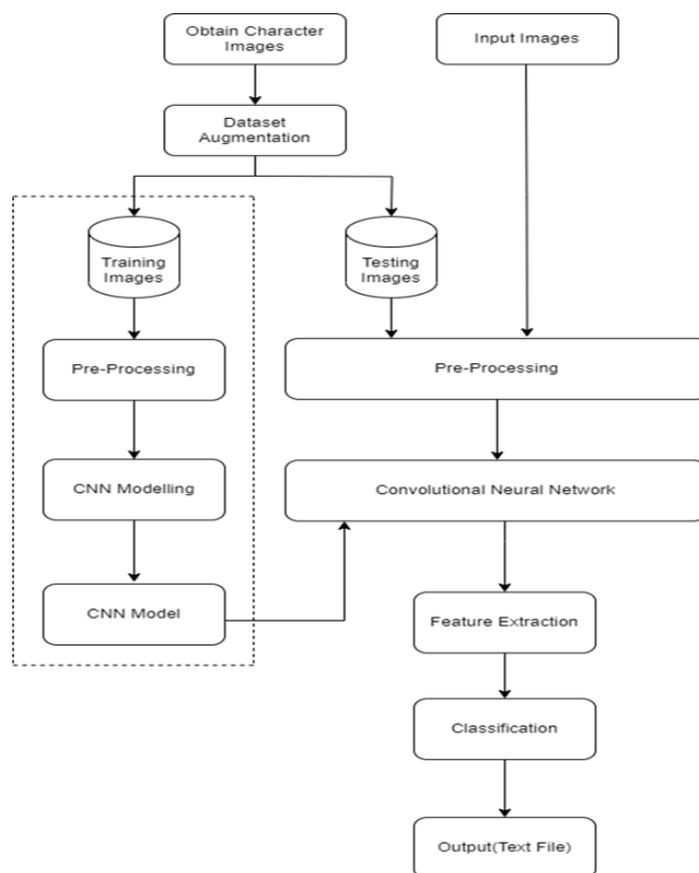


Figure 1. Overall Architecture

2.1. Pre-Processing

Pre-processing is used to remove the undesired elements of the input image and to convert it into a suitable format that makes further processing of the image easier. The images received are of different sizes. Thus, it is necessary to scale them to a suitable size of 86x86. This size is chosen so that it is possible to deepen the network further. If the input size is chosen as the same size as that of LeNet-5, it will be possible to increase the number of convolution operations that takes place inside the network. Most of the images obtained have higher resolution than the size that is needed. Therefore, the image is cropped, selecting only the character field and is scaled to the required size. In case the image is too small, the image is padded with white spaces to make it uniform size.

2.2. CNN Modeling

The most important step is designing and modelling of the CNN. The structure of the CNN is modelled to suit Malayalam handwritten character recognition. The number of layers, type of layers, number of neurons per layer, size of kernel, stride in each operation and the type of classifier to be used are decided. The training dataset is tested against different settings of the network to identify the configuration of the network which produces higher accuracy.

LeNet-5 was the first popular CNN model. It had 3 convolution layers and 2 pooling layers. Its application was limited at the time because of the unavailability of sufficient hardware for processing large training data. However, with the introduction of Compute Unified Device Architecture (CUDA) and parallel processing, a CNN is trained using large amount of data. The dataset used in this method are tested against the LeNet-5 network first.

A CNN model (CNN1) which is smaller than LeNet-5 is built to test how well a CNN performs against the augmented dataset of 6 characters. It has only 2 convolution and 2 max pooling layers. But it has

2 Rectified Linear Unit (ReLU) layers and a fully connected layer as opposed to two in LeNet-5. The input images were of size 86x86. The following Figure 2 shows this first CNN model.

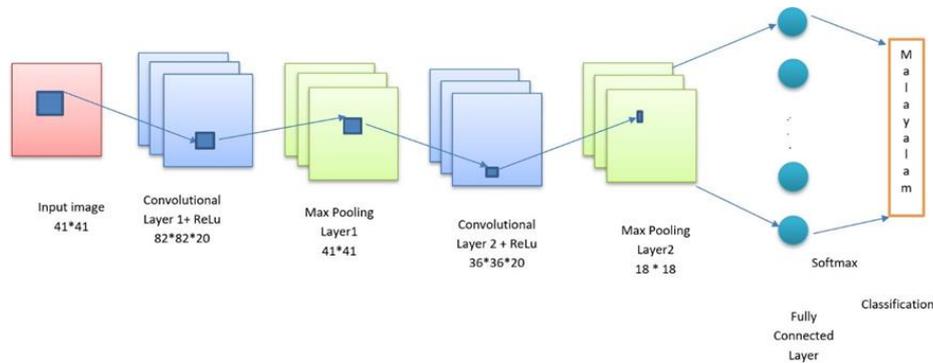


Figure 2. CNN1

Let x be the height of a square image of size $x \times x$ and let y be the height of the convolution layer. Then, the size of the feature map after convolution with kernel will be $x*y+1$, where $y \leq x$. There are different types of sampling techniques for reducing the size of the feature map. Max pooling is used in this method, as it has shown the highest accuracy rate in comparison with other pooling methods. This comparison is discussed in the result section in detail.

The second CNN model (CNN2) designed is shown in the following Figure 3. It has 16 layers including the input layer. The input layer has the input image of size 86x86. The second layer in the network is the first Convolution layer, Conv1. It is obtained by convolving the input image with a 5x5 kernel with stride 1. Six feature maps obtained here is of size 82x82. The next layer in the network is a ReLu layer, ReLu1. The function of ReLu is to eliminate all the negative values that will cause trouble during back-propagation. A rectified linear unit has output 0 if the input is less than 0 and raw output otherwise. That is, if the input is greater than 0, the output is equal to the input. ReLUs' machinery is more like a real neuron in the human body. The max pooling layer MP1 that has kernel 2x2 with stride 2 follows ReLu1 and it reduces the size of the image by half thereby making size of each feature map 41x41.

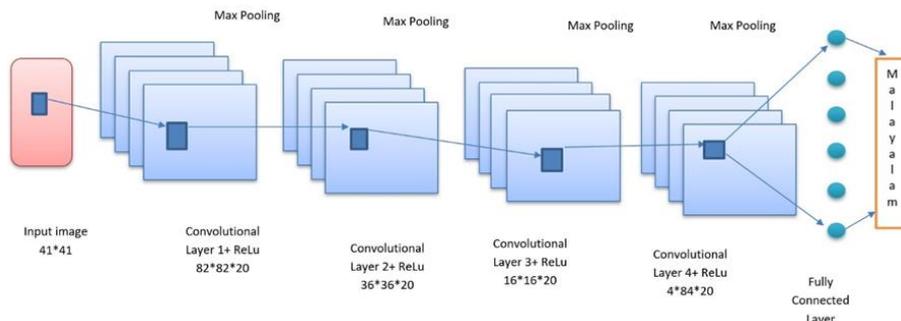


Figure 3. CNN2

The next Convolution Layer, Conv2 follows MP1 and it has a kernel size of 6x6 with stride 1 resulting in 20 feature maps of size 36x36 each. This is followed by ReLu layer, ReLu2. Sub-sampling with kernel 2x2 and stride 2 follows this layer. The sub-sampling method used is max pooling after which there are 20 feature maps with size 18x18 each. Another Convolution layer, Conv3 with kernel size 3x3 and stride 1 gives 100 feature maps of size 16x16. ReLu layer, ReLu3 follows Conv3 and is followed by Max pooling layer, MP3 that reduces the size of each feature map to size 8x8.

The next layer is the fourth and final convolutional layer and it has a kernel of size 5x5 with a stride of 1 which produces 200 feature maps each of size 4x4. A ReLu layer, ReLu4 follows this layer and these feature maps are reduced to size 2x2 by the following max pooling layer MP4 of kernel 2x2 and stride 2. The output of this layer is feed into a fully connected layer having 44 nodes and then into a soft-max layer and subsequently into a classification layer.

The fully connected layer helps in identifying 200 distinct features and is fed into the soft-max layer. The soft-max layer squashes the values received from the previous layer into a value between 0 and 1. The soft-max layer has 44 distinct nodes with each node representing a class. The value obtained in soft-max function is actually a probability that the character formed by the set of feature belongs to a label. The final layer is the classification layer where the input image is classified into any one of the 44 classes depending on the probability values generated by soft-max layer. The learning rule chosen is stochastic gradient descent along with back-propagation.

2.3. Classification

The second last layer of the CNN is a soft-max layer, which has soft-max function for squashing all the inputs into a value between 0 and 1. The sum of all values in a soft-max layer will be between 0 and 1. The next layer is used for classifying the given input image. This classification is based on the values generated by the previous soft-max layer. This layer classifies the input image into one of the 44 classes.

3. RESULTS AND DISCUSSION

The entire dataset is first divided into training and testing dataset. The split is performed on a random basis, so that 80% of the dataset is selected as the training set and the rest is selected as the testing set. This training set is tested on the CNN after training. The class labels of testing images are hidden at first. These images are then passed through the CNN which predicts the class of the input image. Thus, accuracy of the CNN is identified by dividing the number of correct predictions to the total number of test images. Additionally, new raw inputs are considered from different individuals and tested with this network. For this, the input images are first binarized and then scaled to 86x86 size. This image is then feed into the network and the class label of the input image is predicted.

The input picture is created in an editable format after the label of the input character image has been anticipated. The label of each character is used as the index and its Unicode value is stored in a Comma-Separated Values (CSV) file. By referencing this CSV file, the OCR system generates the Unicode from the predicted label.

3.1. Dataset

In this model, 4 different datasets are used. 44 characters are considered from P-ARTS Kayyethu [28] and another raw dataset containing 44 characters is created by taking samples from different people. This raw dataset is further divided into 3 subsets. For easier understanding, these 3 datasets are referred to as *RawDataset1*, for the dataset containing raw images of the first 6 characters; *AugmentedDataset1* for the dataset that was produced by augmenting *RawDataset1*; and *RawDataset2* for the dataset that contains raw images of 44 characters. The size of each character image in these three datasets is 86x86. However, for training and testing LeNet-5 model, the size of the images is reduced to 32x32 for maintaining the originality of the network. *RawDataset1* and *AugmentedDataset1* are used for testing LeNet-5 and CNN1 that is created. The characters from P-ARTS Kayyethu are labelled *CHAR1* through *CHAR44* with a total of 91,902-character images.



Figure 4. Samples of P-ARTS Kayyethu dataset

3.2. System Details

Dell precision tower 5810 workstation with 64 GB RAM and 4 GB DDR5 NVIDIA Quadro m2000 CUDA graphics card is used for this work. The entire work is implemented in MATLAB 2016b. NVIDIA created the parallel computing platform and programming model known as Compute Unified Device Architecture (CUDA).

Accuracy is the criterion for evaluation in this case. Accuracy is further divided into two for this system, namely testing accuracy and accuracy of real-world inputs. In the case of testing accuracy, images that are separated from the dataset and stored as testing set are used. Thus, the testing accuracy is the ratio of images whose predicted label is equal to actual label to the total number of testing images.

$$\text{Testing Accuracy} = \frac{\text{Number of correctly predicted images}}{\text{Total number of input images}} * 100 \quad (1)$$

The accuracy of raw inputs also needs to be checked. It is given by,

$$\text{Accuracy} = \frac{\text{Number of correctly predicted images}}{\text{Total number of images}} * 100 \quad (2)$$

3.3. Results

3.3.1. LeNet-5 vs CNN1

The popular MNIST dataset is used in training LeNet-5 network. The LeNet architecture is trained using four Malayalam handwritten dataset, out of which two contained six characters and the other two contained 44 characters. CNN1 is a simpler network than LeNet-5 architecture and has lesser number of layers than LeNet architecture. While LeNet-5 is trained and tested using images having size 32x32, CNN1 is trained and tested using images of size 86x86. These are tested with *RawDataset1* and *AugmentedDataset1* and the results are shown in Table 1.

Table 1. Comparison of LeNet-5 and CNN1 for six Character dataset

CNN	Accuracy for <i>RawDataset1</i>	Accuracy for <i>Augmented Dataset1</i>
LeNet-5	61.1%	65.4%
CNN-1	79.2%	87.96%

From Table 1, it is clear that the CNN1 shows higher classification accuracy for both the datasets. CNN1 has an accuracy of 79.2% when compared to 61.1% of LeNet-5 in the case of *RawDataset1* and for *AugmentedDataset1*, 87.96% and 65.4% is obtained for CNN1 and LeNet-5 respectively. However, when the dataset containing all 44 characters is tested, the accuracy levels are reduced. Both these models achieved lesser accuracy because the number of classes was significantly higher and this led to over-fitting.

3.3.2. CNN2

The first parameter of CNN2 that is decided is the initial learning rate or the base learning rate. If the base learning rate given is too low, then training takes more time. But, if a high value for initial learning rate is considered, the learning becomes too fast and the network performance decreases. It is necessary to find a base learning rate that produces higher accuracy and so the other parameters like learning rate drop factor, learning rate drop period and max epochs were set to 0.0001, 5 and 10 respectively. Table 2 shows the comparison of AlexNet-24, ResNet and CNN2 for P-ARTS Kayyethuthu dataset. The following Table 3 shows the result of training the same network with different initial learning rates. The following Figure 5 shows a graph that is plotted between base learning rate and accuracy. It is noticed that setting a base learning value of 0.004 gives the highest accuracy of 99.72%.

Table 2. Comparison of AlexNet-24, ResNet and CNN2 for P-ARTS Kayyethuthu dataset

CNN	Accuracy of data set from P-ARTS Kayyethuthu
ResNet	94.65%
AlexNet-24	98.74%
CNN2	99.72%

Table 3. Comparison of accuracy and Training time on varying base learning rate

Base Learning Rate	Accuracy	Training time in seconds
0.001	99.25%	1447.29
0.002	99.53%	1005.35
0.0025	99.68%	1554
0.003	99.512%	956.2
0.0035	99.66%	1016
0.004	99.72%	982.76
0.0045	99.49%	1264
0.005	99.623%	971.72
0.0055	91.11%	1011.39
0.006	99.37%	1407
0.0075	99.47%	962.87
0.008	99.27%	969
0.01	98.26%	959.09

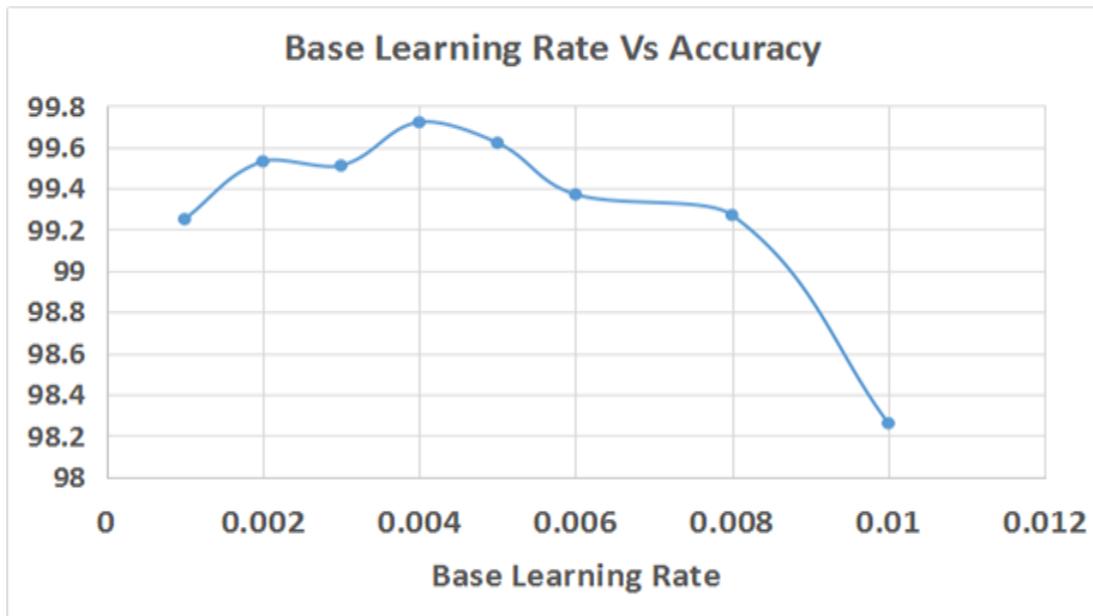


Figure 5. Base learning rate Vs Accuracy

After obtaining the optimal base learning rate, it is necessary to identify an optimal learning rate drop factor. This specifies the drop-in learning rate over a certain number of epochs. From the following Table 4 it is observed that the accuracy increases till the learning rate drop factor reaches 0.05 and then starts to decline. Thus, 0.05 is chosen as the learning rate drop factor for this network. It is noted that the training time drops while learning rate drop period is reduced.

Table 4. Comparison of accuracy and Training time on varying learning rate drop

Learning rate drop	Accuracy	Training time in seconds
0.001	99.72 %	982.79
0.01	99.78 %	1033.88
0.02	99.94 %	1296
0.05	99.95 %	1468
0.1	99.94 %	2402.82
0.5	99.89 %	986.92

Learning rate drop period is defined as the rate at which the learning rate is changed. Generally, these periods are epochs and different learning rate drop periods are checked. The drop period is checked with different values and maximum accuracy is obtained when the number of epochs is 5. So, the number of epochs is set to 5 for testing. The results are shown in Table 5.

Table 5. Comparison of accuracy and training time on learning rate drop period

Learning rate drop period	Accuracy	Training time in seconds
1	94.06 %	1447.29
2	99.58 %	672.4
3	99.90 %	1554
4	99.94 %	956.2
5	99.95 %	1016
10	99.75 %	608.2

A bigger CNN (CNN2) having 16 layers is built. Different parameter values and different techniques for pooling are included in the modified network which consists of four convolutional blocks. As well as slight modification in the network are carried out. Max pooling and average pooling are two of the popular sampling methods. Both the sampling methods are tried on the CNN2 network. The results are shown in the following Table 6. While there is an increase in the accuracy for max pooling over average pooling, there is also a slight increase in the training time as well.

Table 6. Comparison of Pooling strategies

Pooling Method	Testing Accuracy	Training time in seconds
Average pooling	99.759%	946.96
Max pooling	99.955%	1418.4

The number of feature maps that is generated after each convolution is varied and the results are summarized in the following Table 7. When the number of features taken is less, the accuracy is decreased. But, if the number of feature maps is increased, it is identified that the accuracy increases along with training time. Increasing the feature map above a certain limit, results in a marginal increase in accuracy. At the same time there is a large increase in the training time. When the number of feature maps is further increased, a

memory out of bound error occurs. This is because the computation of such sort requires even higher memory and a greater number of GPU devices. Thus, it is not possible to estimate the accuracy and training time of this combination of feature map values.

Table 7. Comparison of change in number of feature maps for CNN2

Class1	Class2	Class3	Class4	Accuracy	Training time in seconds
10	20	30	50	99.79%	369.13
20	20	50	100	99.95%	1479
20	20	50	75	99.81%	1518
100	150	100	75	-	-
20	20	100	200	99.96%	1155

All the training samples from the training set forms a batch. Since Stochastic Gradient Descent (SDG) is used, the mini-batch size is calculated at each step. The mini-batch size is varied from 100 to 1000 and obtained the highest accuracy of 99.96% for mini-batch size 300. The results are summarized in the following Table 8.

Table 8. Comparison of accuracy and training time on varying the mini-batch size

Mini-batch size	Accuracy	Training time in seconds
100	99.93%	939.9
300	99.96%	1155
500	99.85%	1290
1000	99.68%	1814.64

The training accuracy of CNN2 is shown in the following Fig. 6. After a certain number of iterations the training accuracy tends to stay above 98%. The training accuracy climbs high during the initial phase of training and tends to remain unchanged after training the network over a long time.

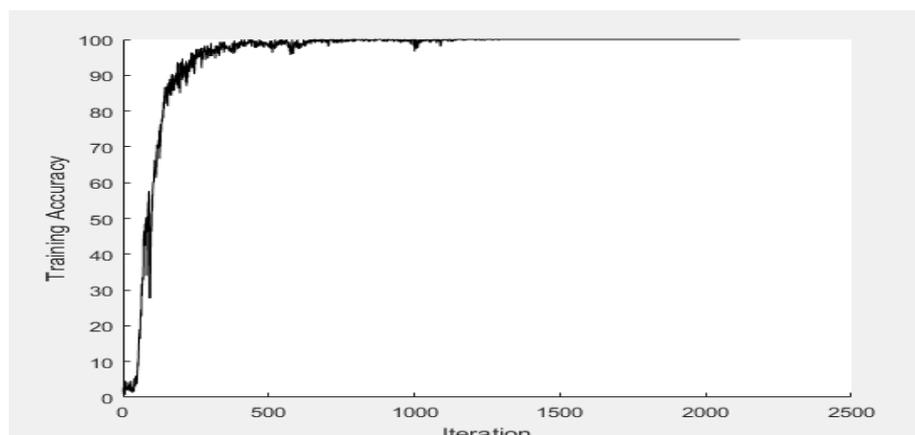


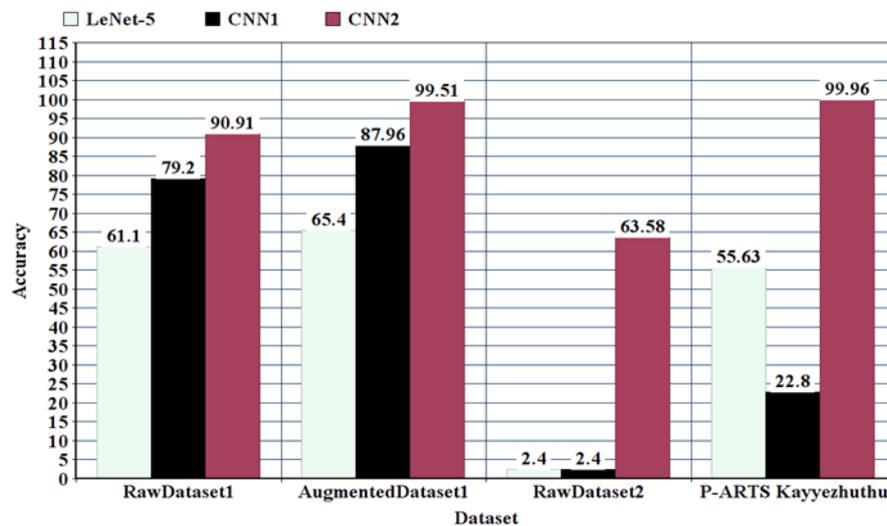
Figure 6. Training accuracy vs number of iterations

From the Table 9 it is observed that, CNN2 outperforms the other two networks in the case of all 4 datasets and hence, CNN2 is the best CNN among the three models. CNN2 has an accuracy of 90.91% for *RawDataset1*, 99.51% for *AugmentedDataset1* and 99.96% for P-ARTS Kayyvezhuthu. A graph of this

comparison is shown in the following Fig. 6. It is noticed that CNN1 has higher accuracy than LeNet-5 configuration in case of *RawDataset1* and *AugmentedDataset1*. However CNN1 did not exhibit the same consistency with *RawDataset2* and P-ARTS Kayyehzuthu.

Table 9. Comparison of LeNet-5, CNN1, and CNN2 using all 4 datasets

CNN	Accuracy for <i>Raw Dataset1</i>	Accuracy for <i>Augmented Dataset1</i>	Accuracy for <i>Raw Dataset2</i>	Accuracy for P-ARTS Kayyehzuthu
LeNet-5	61.1%	65.4%	2.4%	55.63%
CNN1	79.2%	87.96%	2.4%	22.8%
CNN2	90.91%	99.51%	63.58%	99.96%



4. CONCLUSION

Optical character recognition finds application in many days to day activities that include number plate recognition and office automation. Convolutional Neural Networks have not been implemented for handwritten Malayalam Characters yet. A new OCR using Convolutional Neural Network for Malayalam handwritten character recognition is implemented here.

The system does feature extraction by itself without user-defined features and thus removes the need for handcrafted features that need to be developed by the programmer. A new dataset was created, and it is made available publicly for further research. The CNN was tested using the same, and an accuracy of 99.96% was obtained. Different configurations of the network were tested. LeNet-5 Architecture was compared against a smaller network CNN1 which had 2 ReLU layers and performed better for *AugmentedDataset1*. The network parameters were varied and the best combination for getting high accuracy and low training time was chosen as CNN2. CNN2 provides an accurate model for handwritten character recognition. External inputs were also given to the system and it was found that for 40 such input images an accuracy of 95% was obtained. However, the system can be improved by collecting even more images and expanding the dataset. By collecting more images for training the system can be made to identify more variations to a character. The dataset can also be expanded to include all the special characters and other combination characters.

ACKNOWLEDGMENTS

You may want to thank your funding source (but do not thank to any of the authors!).

REFERENCES

- [1] Asifullah Khan , Anabia Sohail, Umme Zahoor , and Aqsa Saeed Qureshi "A Survey of the Recent Architectures of Deep Convolutional Neural Networks", Published in Artificial Intelligence Review, DOI: <https://doi.org/10.1007/s10462-020-09825-6>.

- [2] [Bindu Madhuri Cheekati, Roje Spandana Rajeti](#) (2020) "Telugu handwritten character recognition using deep residual learning", [Fourth International Conference on I-SMAC \(IoT in Social, Mobile, Analytics and Cloud\) \(I-SMAC\)](#).
- [3] [G. Ramesh, Sandeep Kumar N., H.N. Champa](#), (2020) "Recognition of Kannada Handwritten Words using SVM Classifier with Convolutional Neural Network", [2020 IEEE Region 10 Symposium \(TENSymp\)](#).
- [4] Manjusha J, Ajay James, C Sharavanan, (2018) "Malayalam Handwritten Character Recognition Using AlexNet Based Architecture", *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)* Vol. 6, No. 4, December 2018, pp. 393~400 ISSN: 2089-3272, DOI: 10.11591/ijeei.v6i1.518.
- [5] Samatha P Salim, Ajay James, Chandra Saravnan (2019) "Proposed method to Malayalam Handwritten Character Recognition using Residual Network enhanced by multi-scaled features" 1st IEEE International Conference on Innovations in Information and Communication Technology.
- [6] Bhagyasree P V, Ajay James, Chandra Saravnan (2019) "A Proposed Framework for Recognition of Handwritten Cursive English Characters using DAG-CNN" 1st IEEE International Conference on Innovations in Information and Communication Technology.
- [7] Yasin Zamani, YaserSouri, Hossein Rashidi and Shohreh Kasaei. (2015). Persian Handwritten Digit Recognition by Random Forest and Convolutional Neural Networks ,*9th Iranian Conference on Machine Vision and Image Processing*, (November 18-19,2015;Shahid Beheshti University -Tehran, Iran.).
- [8] Shailesh Acharya, Ashok Kumar Pant, Prashanna Kumar Gyawali. (2015). "Deep Learning Based Large Scale Handwritten Devanagari Character Recognition",*9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*.
- [9] Jomy John, Pramod K. V, KannanBalakrishnan.(2011) "Unconstrained Handwritten Malayalam Character Recognition using Wavelet Transform and Support vector Machine Classifier", *International Conference on Communication Technology and System Design*.
- [10] Abdul Rahiman M and Rajasree M S. (2011) "An Efficient Character Recognition System for Handwritten Malayalam Characters Based on Intensity Variations" ,*International Journal of Computer Theory and Engineering, Vol. 3, June*.
- [11] Manju Manuel and Saidas S. R. (2015) "Handwritten Malayalam Character Recognition using Curvelet Transform and ANN", *International Journal of Computer Applications (0975 8887) Volume 121 No.6, July*.
- [12] Meiyin Wu and Li Chen .(2015). "Image Recognition Based on Deep Learning",*Chinese Automation Congress (CAC)*.
- [13] Yujie Liu and He Huang. (2015), "Car plate character recognition using a convolutional neural network with shared hidden layers", *Chinese Automation Congress (CAC), 2015*.
- [14] Prashanth Vijayaraghavan and Misha "Handwritten Tamil Recognition using a Convolutional Neural Network", *MIT Media Lab*.
- [15] Anitha Mary M.O. Chackoa, Dhanya P.M. (2014). "Multiple Classifier System for Offline Malayalam Character Recognition", *International Conference on Information and Communication Technologies (ICICT)*.
- [16] G. Raju, Bindu S Moni, Madhu S. Nair. (2014) "A Novel Handwritten Character Recognition System Using Gradient Based Features and Run Length Count",*Sadhana Indian Academy of Sciences, Springer India, 2014, p. 1-23*.
- [17] AroraSandhya, Bhattacharjee Debotosh, NasipuriMita, BasuDipak Kumar and KunduMahantapas. (2008)*Third international Conference on Industrial and Information Systems ICIIIS, 2008, p. 1-6*.
- [18] Ajay James, Raveena P.V and Chandran Saravanan, "Handwritten Malayalam Character Recognition using Regional Zoning and Structural Features", *International Journal of Engineering & Technology, 7 (4) (2018) 4629-4636 International Journal of Engineering & Technology Website: www.sciencepubco.com/index.php/IJET doi: 10.14419/ijet.v7i4.12551, 2018*.
- [19] Jino P J ,KannanBalakrishnan, "Offline Handwritten Recognition of Malayalam District Name - A Holistic Approach", *arxiv:1705.00794*.
- [20] Ajay James, Sonu Varghese K and Dr. Chandran Saravanan, "Hybrid Approach of Handwritten Malayalam Character Recognition using Geometrical and Structural Properties", *International Journal of Engineering & Technology, 7 (4) (2018) 4624-xxx International Journal of Engineering & Technology Website: www.sciencepubco.com/index.php/IJET doi: 10.14419/ijet.v7i4.11994, 2018*.
- [21] Jomy John, KannanBalakrishnan, Pramod K. V. (2013) "A System for Offline Recognition of Handwritten Characters in Malayalam Script", *I.J. Image, Graphics and Signal Processing, Volume no:4 .pg:53-59, 2013*.
- [22] Meenu Alexa, Smija Das. (2016) "An Approach towards Malayalam Handwriting Recognition Using Dissimilar Classifiers",*Procedia Technology 25 224 231*.
- [23] Jomy John, Kannan Balakrishnan, Pramod K. V. (2016). "Handwritten Malayalam character recognition with a novel gradient based feature descriptor and a comparative study using SVM and ELM", *International Journal on Advances in Engineering Technology and Science, 2, Issue: 3, May-June*.
- [24] Nishad A, Dr. Bindu K (2013) "Malayalam OCR Systems A Survey",*International Journal of Computer Technology and Electronics Engineering (IJCTEE) Volume 3, Issue 6, December*.
- [25] Jomy John, Pramod K. V, KannanBalakrishnan. (2011) "Offline Handwritten Malayalam Character Recognition Based on Chain Code Histogram", *International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT)*.

- [26] Shanjana C, Ajay James. (2014). "Offline Recognition of Malayalam Handwritten Text", *8th International Conference on Interdisciplinary in Engineering, INTER-ENG*, Romania.
- [27] Li Chen, Song Wang, Wei Fan, Jun Sun, Satoshi Naoi. (2015). "Beyond Human Recognition: A CNN-Based Framework for Handwritten Character Recognition", *3rd IAPR Asian Conference on Pattern Recognition*, 2015.
- [28] P-ARTSKayyethuthu <https://drive.google.com/drive/folders/0B1eLyjUeuERZWUVIOU9OZm40RHc>
- [29] <https://en.wikipedia.org/wiki/Malayalam>

BIOGRAPHY OF AUTHORS



Ajay James completed his Bachelor of Engineering from N I College of Engineering and M Tech from Pondicherry University. He has completed his PhD from NIT Durgapur. He has more than 17 years teaching experience in various engineering colleges and is currently working as Associate Professor in Government Engineering College Thrissur.



Philomina Simon is currently working as Assistant Professor, Department of Computer Science, University of Kerala. She completed her M. Tech from Department of Computer Science at Pondicherry Central University in 2009 and B.Tech in Information Technology from University of Kerala in 2005. She has a teaching experience of 14 years. She also worked as Assistant Systems Engineer in Tata Consultancy Services (TCS), Mumbai. She is a member of various professional bodies such as IAENG, CSI, IACSIT. She served as a board member in various universities. She has research publications in several International journals and Conferences. Her research interests include Image Processing, Computer Vision, Machine Learning, Deep Neural Networks.



Pranav Nair completed his Bachelors in computer science and engineering from Coorg institute of Technology, Ponnampet in 2014 and his Masters in computer science and engineering from Government engineering college Thrissur in 2017. He has over 5 years of experience working in the IT industry and is now working as a data engineer in an MNC.



Bhagyasree P V completed her Bachelors in Computer Science and Engineering from Sahrdaya College of Engineering and Technology, Kodakara in 2016 and Masters in Computer Science and Engineering from Government Engineering College Thrissur in 2019. She has 3 years of teaching experience and currently working as Assistant Professor in Christ College of Engineering, Irinjalakuda, Thrissur.