Nat. Volatiles & Essent. Oils, 2021; 8(5): 164 - 175

A Cnn Model Based Approach For Offline Handwritten Tamil Text Recognition System

R.C. Suganthe[1], Pavithra K[2], N. Shanthi[1], R.S. Latha[1]

[1,]Department of Computer science and Engineering, Kongu Engineering College, India,

[2] PG-Scholar, Department of Computer science and Engineering, Kongu Engineering College, India.

Suganthe_rc@kongu.ac.in, pavithrakolandasamy.97@gmail.com

ABSTRACT

Computers may dominate our lives but using pen for writing is still mightier than keyboard. Computer vision applications like handwriting recognition model is playing a vital role now-a-days. Handwritten text recognition (HTR) is the most efficient way to digitize handwritten documents. For every handwritten text recognition model, feature extraction is the foremost task. In CNN architecture, feature extraction is done automatically. Lot of attention has received in the past years but research has focused only on Latin, Urdu and English. For Tamil language very fewer studies were done. Instead of predicting by word, first the individual characters from the text should be segregated then the segregated character is given to the trained CNN model to predict handwritten Tamil characters.The dataset used here is an isolated handwritten Tamil character dataset which was developed by HP labs India. Hyper-parameter tuning is performed to tune the model and achieved good accuracy results.

Keywords: Computer vision, Handwritten text recognition, Hyper-parameter tuning, CNN

1. INTRODUCTION

The oldest living language, Tamil is declared as a classical language by UNESCO in 2004. Tamil Brahminin scriptions found in Adichanallur proved that it has been born before 500 BC. It is predominantly used in Sri Lanka, South India and Malaysia. Every word pronounced in Tamil has a syllable; this is the specialty of Tamil language. In Tamil, for representing a word only minimal syllable is required.Syllable is the smallest unit. Tamil script syllabic units have a unique character (AyudhaEzhuthu, ஃ ,(twelve vowels and eighteen consonants. There are 216 compound letters, from vowels and consonants combination a sum of 247 characters were generated. Also, Tamil have borrowed 5 consonants from Sanskrit language, which would produce another 60 compound characters when combined with vowels. Totally Tamil has 307 charcters. The combinations of 156 distinct characters can depict the entire Tamil character set.

The hardest aspect about handwritten text recognition is the distinctions of human style of writing. Still, it is a challenging task to identify TAMIL characters due to distortions, disposition, and different character length, wide variety of handwriting, fragmented strokes and noises. A standard method of HTR systems includes pre-processing of raw data, segmentation, extraction of important features and finally the classification. Handwritten text recognition systems can be used in various fields; some of them are bank cheque interpretation, Postal service mail sorting, as well as conversion of handwritten documents into digitized documents.

Like every handwritten text recognition process, HTR model for Tamil characters also includes the following preprocessing steps; conversion of colored image into grayscale image , then grayscale image to binary image ,from the background extracting the foreground character , performing morphological transformation to remove noises , line segmentation and then character segmentation .After preprocessing the segmented individual characters are fed to the trained CNN model to extract the features and to predict the character. The objective is to develop a CNN model for handwritten text recognition in Tamil language.

Lots of datasets are available for Chinese, Arabic, Urdu languages. For Tamil, there is an isolated handwritten character dataset named hp-labs-tamil-iso-character. The hp-labs is located at Bangalore. Table 1 will show the detailed description of dataset. The associated works in the handwritten text recognition model are discussed in the section 2. The complete research work is discussed in next section 3. The experimental results and conclusion are discussed in section 4 and 5 respectively.

Table 1.Dataset description

Dataset	**HPLabs India**
Training set	50,683 images
Testing set	26,926 images
Validation set	10,137 images
Total	87,746 images

2. RELATED WORK

In recent year handwriting recognition model plays a vital role in deep learning techniques. Even though lots of researches have been done for other languages, handwriting recognition model for Tamil language is still a challenging one. Kavitha B.R and Srimathi C in [1] have developed CNN model for predicting handwritten Tamil character and they achieved 95.16% training accuracy and 92.74% validation accuracy. For handwritten Devanagari digits Hanmandlu et al developed fuzzy based approach[2], which achieved 95% accuracy. Debabratasenapati, Sasmita rout and Mamata nayak proposed a method for segmenting printed odisa character [3]. Correctness of segmented characteris mandatory for any handwriting recognition model. They achieved 99% and 86% accuracy for segmenting the lines and words respectively.

Prashanth vijayaraghavan and Mishasra have used CovNets for feature extraction along with probabilistic weighted pooling, for IWFHR-10 dataset they achieved 94.4% accuracy. The SVM-based recognition algorithm used by Shanthi and Duraisamy yielded remarkably accurate results[5]. To extract the required features from a character image, zoning has been applied and pixel density values have been generated. Chauoki Boufenar, Adlenkerboua and Mohamed Batouche developed CNN model [6] for predicting Arabic characters via transfer learning achieved 100% accuracy. Also they tested the model on two datasets. In 2021 [4] vinotheni, lakshmi proposed modified convolutional neural network (M-CNN) architecture for predicting the Tamil character and achieved 97.07% accuracy.

Amitabh Wahi and Tiji M developed wavelet transformation technique [7] for extracting the salient features of handwritten Tamil character and feed forward backpropagation technique for classification. Nina Aleskerova and AlekseiZhuravlev used a hierarchical multiple neural network[8], in which the first level neural network selects the second level neural network, which then performs the actual recognition. Nibaran Das, Riya Guha, Mahantapas, MitaNasipuri and Santosh have developed CNN model for predicting handwritten Devanagari characters[9]. They made a comparative study on, DenseNet, ResNet, InceptionNet v3, LeNet−5 etc,

Zhang et al combined CNN with normalization and achieved 97.37% in offline mode[10]. For segmenting the line from the text Darko Brodi´c has used water flow algorithm[11]. For many optical character recognition systems[12], the crucial step is line segmentation. It's a labeling method that assigns spatially aligned units like linked elements, pixels, or characteristic points with same mark[13].

KarthiM, Priscilla R, Syed Jafer K [14] used Convolutional Nueral Network to predict handwritten characters and increased the rate of prediction. E.Sreenivasa Reddy, Ch.N.Manisha, and Y.K.Sundara in [15] described seven applications based on offline handwritten characters recognition system and gives the importantace of HCR systems in various application. The characters in ancient documents are very difficult to predict due to damages Alfons Juan et. al. proposed a method to predict characters from ancient documents[16-18].

3. MATERIALS AND METHODOLOGY

Digitization of handwritten text recognition model including following steps:

1) Developing CNN model.
2) Performing hyper parameter tuning.
3) Implementing character segmentation algorithm.
4) Recognizing the text by giving character one by one to the tuned model.

The overall architecture of Tamil handwritten text recognition system by using CNN architecture in offline mode is shown in figure 3.1 .

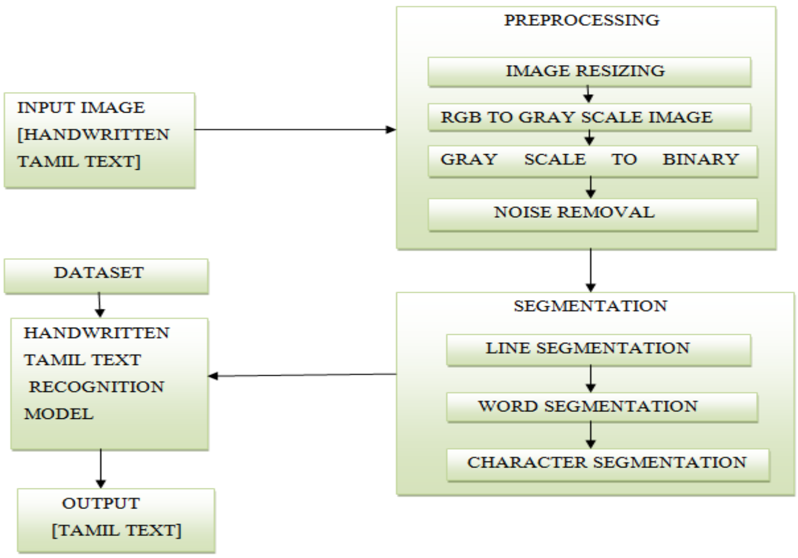

Figure 3.1 Offline handwritten Tamil text recognition model architecture

3.1 THE CNN MODEL

Convolutional neural network is mainly used for image classification because of its accuracy and automatic feature extraction. The raw input image to be given to the model should be preprocessed to get better accuracy. The CNN can be also used for video recognition, medical image analysis and natural language processing. 1-dimensional CNN is used for text classification and 2-dimensional CNN is used for audio and image classification. The main advantage of CNN is that it does not require feature engineering and there is no need to spend lots of time for feature selection. The image classification model requires large amount of datasets, so CNN architecture is suitable for such cases.

The segmented Tamil handwritten character image is given as an input to convolutional layer. The entire image is broken into tiny areas and then, weights and bias are implemented over that. Filter is the one which identifies the spatial patterns that present in input images by detecting the shift in image intensity.The filter size used in convolutional layer is about 3 * 3 with stride 1. The number of kernel, kernel size, stride value and padding value are the convolutional layer hyper parameters. The segmented Tamil character image size is about 64*64. Based upon the size of input image these hyper parameters are chosen.The filter size that we used in convolutional layer is about 3 * 3 with stride 1 and padding 1.

The purpose of pooling layer in HTTR [handwritten Tamil text recognition] system is to decrease the spatial dimension of the segmented Tamil character without adding parameters to the model. Adding more layers may leads to loss of features which will impact accuracy. Added only three limited number of pooling layer, this will helps to retain important features. Max pooling of filter size 2*2 is used.This layer offers translational invariance, which means that the picture can be recognized even if the location changes slightly.

The extracted features of handwritten Tamil character image are given as an input to fully connected layer. Since Hp-labs-Tamil-iso-character-dataset contains 156 unique characters, there are 156 neurons in fully connected layer. Among 156 neurons, the one which have higher probability will be fired by activation function. That fired neuron will map the correct classification label of given handwritten Tamil character segmented input image[19-22].

The output of a neural network is determined by activation functions, which are mathematical equations. Activation function decides whether to activate the neuron or not based on the input given. The activation function used is Relu. Relu (rectified linear unit) will return zero if the input value is negative and returns the value itself if it is positive. The CNN model architecture is shown in diagram 3.1.

3.2. TAMIL CHARACTER SEGMENTATION

The steps involved in handwritten character segmentation process are preprocessing, linesegmentation, word segmentation and character segmentation. Figure 3.2 will show the flow of Tamil handwritten character segmentation process.

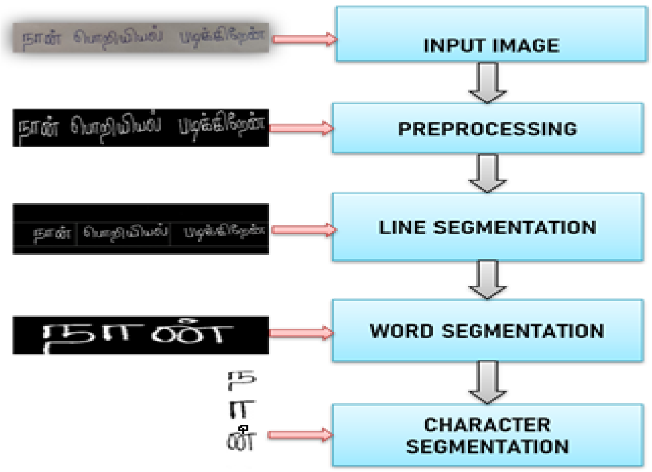

Figure 3.2. Segmentation process

The raw input image should be preprocessed before giving to the developed CNN model. This enhances the important features and remove the unwanted noises. First all the images are resized into uniform size and coverted to grayscale image. After that morphological transformation is applied to remove the noise is present in the binary image. The preprocessed image is further processed to segement the individual lines from the handwritten Tamil text image. The line segmentation counts all the projection profile of the source image along horizontal axis for every row and find all the starting line and ending line for each and every line of the source image and refine it to remove unwanted lines.

From the segmented line image, individual words are segregated for further processing. The contour method is used to get the total number of contours from the segmentedline. Using bounding rectangle method, the average width of the characters is counted and projection profile of the source image along the vertical axis for every column is counted. To remove the unwanted lines, projection profile is refined. After segmenting each word from the line, characters in each word are segmented. The Handwritten Tamil text recognition model takes individual Tamil character image as an input.

4. RESULTS AND DISCUSSION

Learning rate, no of epochs, batch size, activation functions, weight initialization are the hyper parameter tuning variable of convolutional neural network. The various hyper parameters used to fine tune the model is described in table 2.

HYPER PARAMETRS	VALUES
Input image size	64 x 64
Convolution Filter	3x3
Pooling	Max Polling
Pooling window size	2 x 2

Padding	Valid padding
Stride	1 x 1
Optimizers	Stochastic gradient descent (SGD), RMS Prop, ADA Grad
Learning Rate	0.0001, 0.001, 0.01, 0.1, 0.2, 0.3
Activation function	Tanh, ReLU
Weightinitialization algorithm	Kaiming, Xavier

Table.2 Hyper parameter description

The model is experimented with 57 different combinations of hyper parameters. Some of them are listed below in the table 3.Also the accuracy graph is shown in figure.4.1. The epoch size is fixed as 100 and the batch size is fixed as 64.

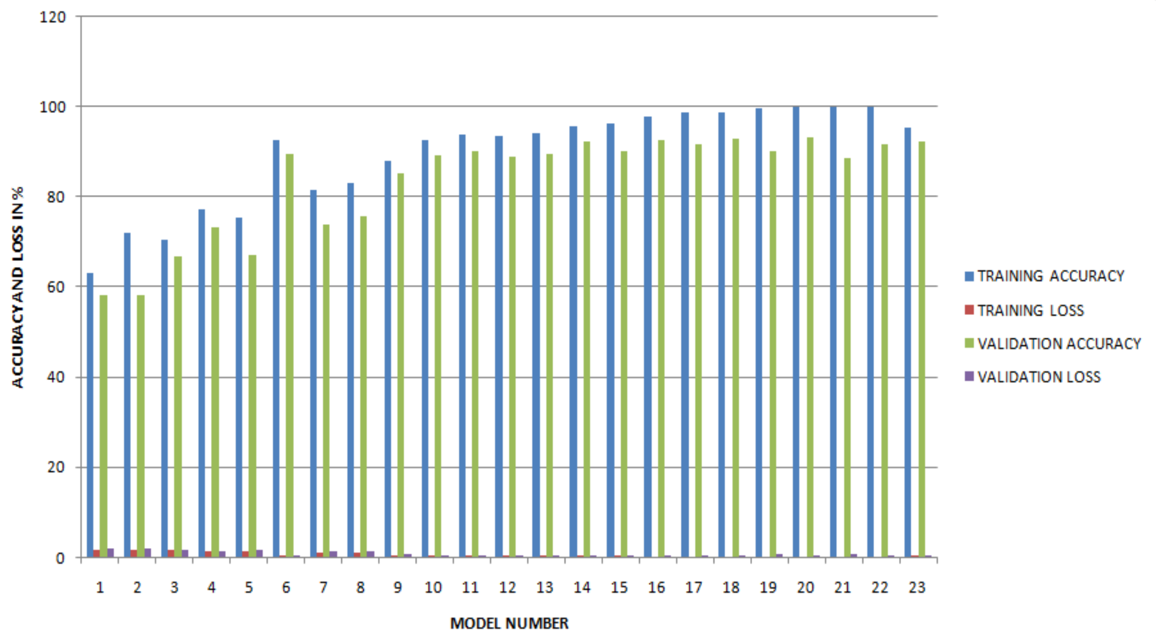

Figure 4.1 Hyper parameter tuning accuracy graph

MODEL NO	OPTIMIZERS	LEARNING RATE	WEIGHT INITIALIZATION	ACTIVATION FUNCTION	TRAINING ACCURACY	TRAINING LOSS	VALIDATION ACCURAC	VALIDATION LOSS
1	SGD	0.0001	KAIMING_NORMAL	Tanh	62.96	1.72	58.22	1.86
2	SGD	0.0001	KAIMING_NORMAL	Relu	71.87	1.73	58.22	1.87
3	SGD	0.0001	XAVIER_NORMAL	Tanh	70.43	1.49	66.74	1.61
4	SGD	0.0001	XAVIER_NORMAL	Relu	77.08	1.23	73.2	1.33
5	ADA_GRAD	0.0001	XAVIER_NORMAL	Tanh	75.35	1.24	67.22	1.47
6	ADA_GRAD	0.2	XAVIER_NORMAL	Tanh	92.72	0.36	89.59	0.44
7	ADA_GRAD	0.0001	KAIMING_NORMAL	Relu	81.44	1	73.7	1.22
8	ADA_GRAD	0.0001	XAVIER_NORMAL	Relu	82.95	0.98	75.72	1.18
9	ADA_GRAD	0.3	KAIMING_NORMAL	Tanh	87.95	0.5	85.3	0.59
10	ADA_GRAD	0.2	KAIMING_NORMAL	Tanh	92.47	0.34	89.18	0.44
11	ADA_GRAD	0.2	KAIMING_NORMAL	Relu	93.94	0.3	90.24	0.4
12	SGD	0.3	KAIMING_NORMAL	Tanh	93.6	0.29	88.83	0.44
13	SGD	0.3	KAIMING_NORMAL	Relu	94.21	0.26	89.58	0.43
14	RMS_PROP	0.001	XAVIER_NORMAL	Relu	95.64	0.23	92.41	0.33
15	ADA_GRAD	0.1	KAIMING_NORMAL	Tanh	96.25	0.22	90.3	0.39
16	ADA_GRAD	0.1	KAIMING_NORMAL	Relu	97.88	0.16	92.63	0.33
17	SGD	0.1	KAIMING_NORMAL	Relu	98.7	0.12	91.82	0.39
18	SGD	0.1	XAVIER_NORMAL	Relu	98.91	0.11	92.91	0.34
19	RMS_PROP	0.0001	KAIMING_NORMAL	Tanh	99.84	0.08	90.23	0.49
20	RMS_PROP	0.0001	KAIMING_NORMAL	Relu	99.89	0.07	93.13	0.38
21	ADA_GRAD	0.001	XAVIER_NORMAL	Tanh	100	0.07	88.68	0.54
22	ADA_GRAD	0.001	XAVIER_NORMAL	Relu	100	0.06	91.71	0.42
23	RMS_PROP	0.001	KAIMING_NORMAL	Relu	95.52	0.22	92.16	0.39

Table 3. Performance of various models.

Two weight initialization algorithms are used to fine tune the model :Kaiming weight initialization algorithm and Xavier weight initialization algorithm. Performance of the model with these two weight initialization algorithms are analysed by varying learning rates such as 0.0001, 0.001, 0.01, 0.1, 0.2 and 0.3. The figure 4.2 illustrates the accuracy of the model with kaiming weight initialization algorithm and figure 4.3 illustrates the accuracy graph of the model with Xavier algorithm. Number of epochs 100, optimizer is SGD, batch-size used is 64, and activation function is ReLU. These parameters are fixed for all the combinations.

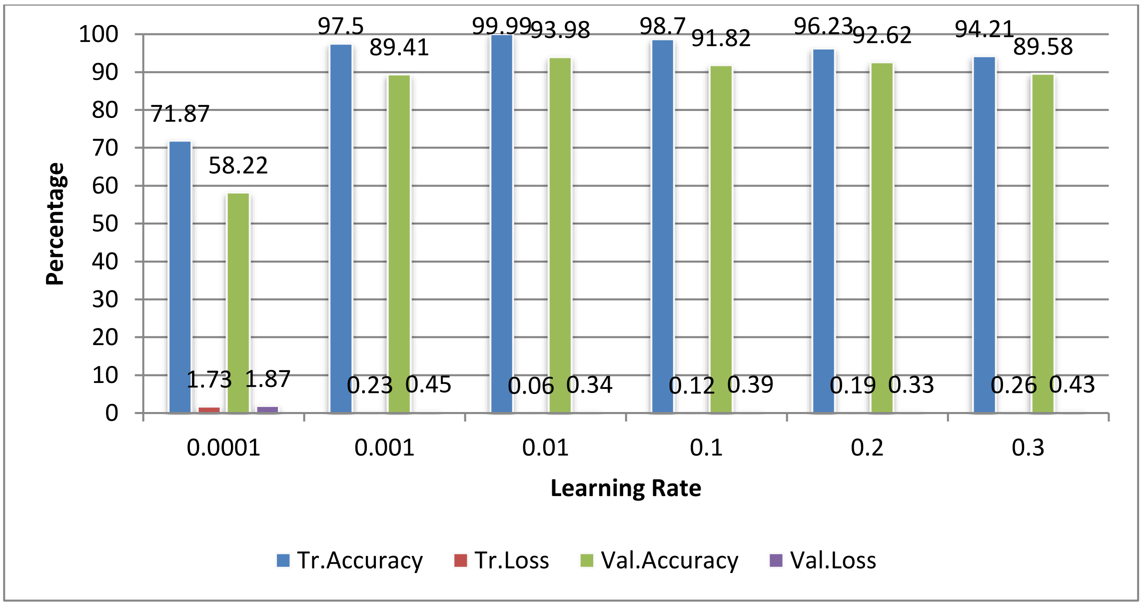

Figure 4.2 Performance of the model with varing learning rate and kaiming weight initialization algorithm

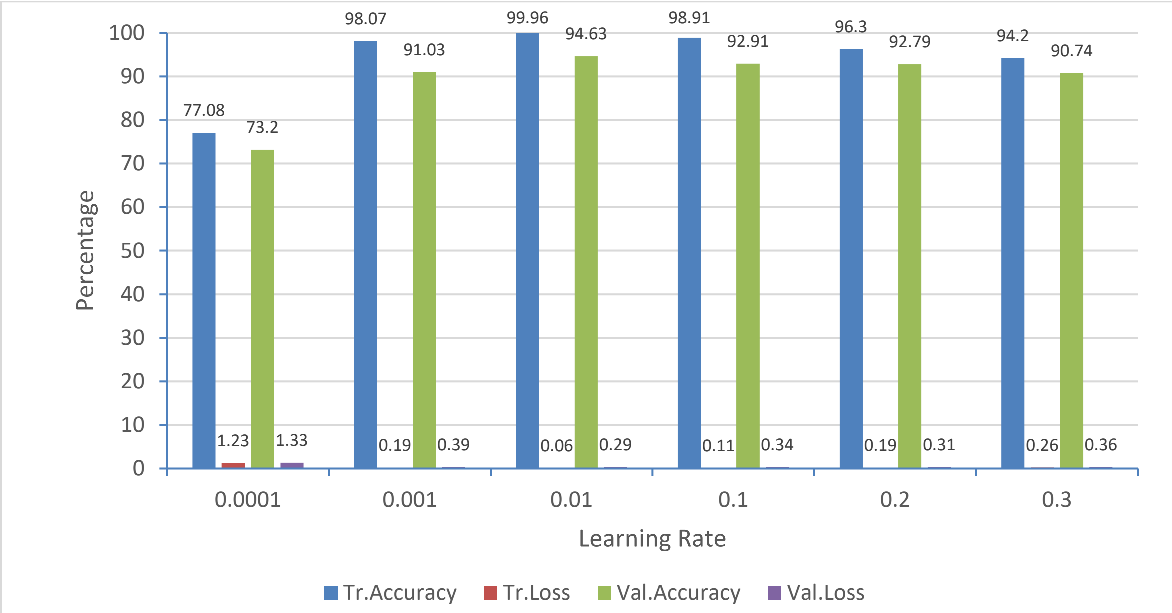

Figure 4.3 Performance of the model with various learning rate and Xavier weight initialization algorithm

When a non linearity function ReLU is used, Xavier algorithm works better than the kaiming algorithm. From the above two graphs it is been inferred that Xavier works well for model that uses non linearity activation functions. In figure 4.4 shows the performance of model for various optimizers such as stochastic gradient descent, RMS Prop, ADA Grad. This model is set with batch size 64, activation function ReLu, kaiming initialization algorithm and learning rate 0.001 and it is trained for 100 epochs. The figure 4.5 shows the performance of model with same parameters with Xavier weight initialization algorithm.

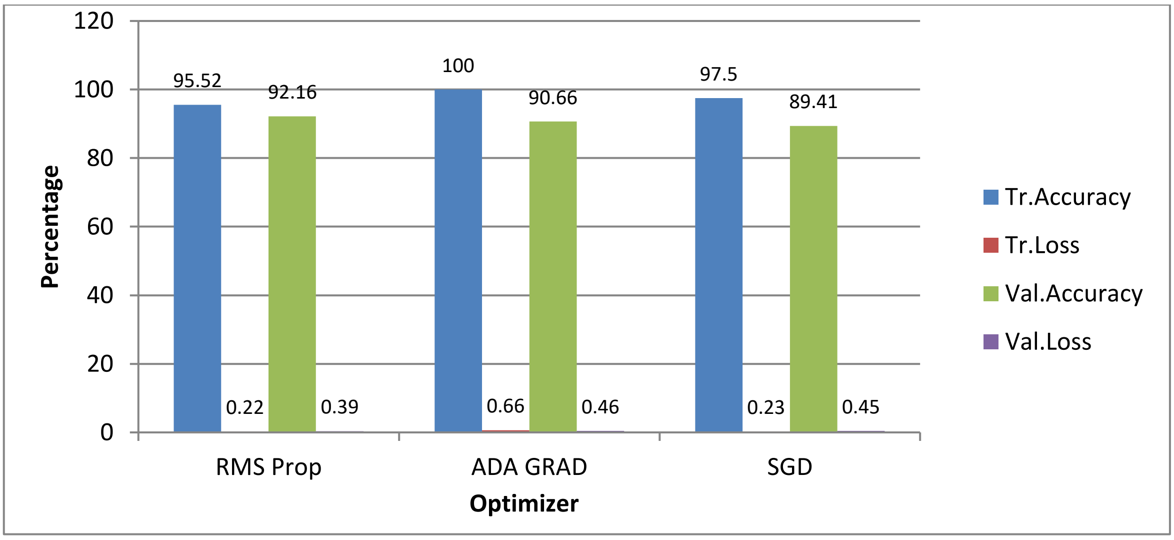

Figure 4.4. Performance of the model with learning rate 0.001 and kaiming weight initialization algorithm

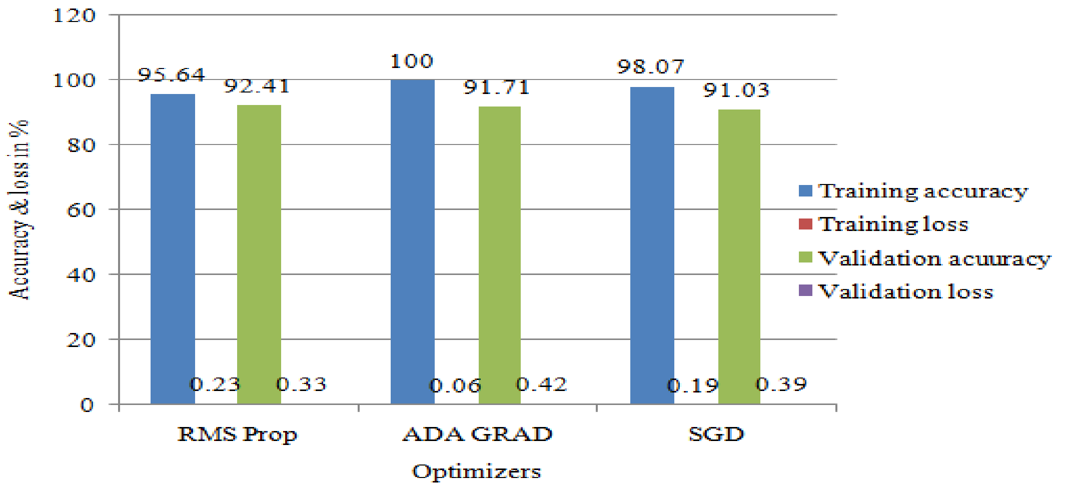

Figure 4.5. Performance of the model with learning rate 0.001 and Xavier weight initialization algorithm

The figure 4.6 show the performance of model for various optimizers such as stochastic gradient descent, RMS Prop, ADA Grad. This model is set with batch size 64, activation function ReLu, kaiming initialization algorithm and learning rate 0.1 and it is trained for 100 epochs. The figure 3.9 shows the performance of model with same parameters with Xavier weight initialization algorithm.

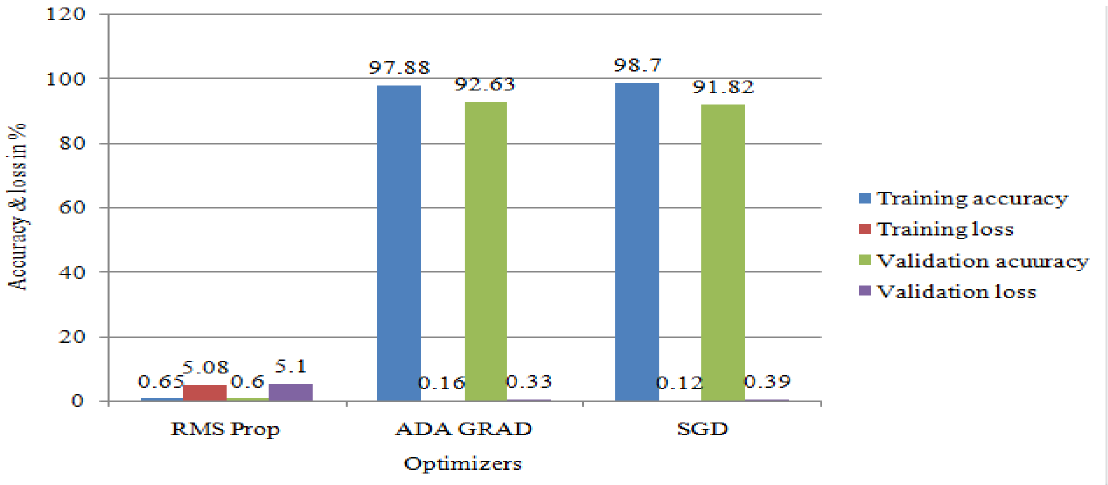

Figure 4.6. Performance of the model with learning rate 0.1 and Xavier weight initialization algorithm

Figure 4.7 shows the performance of model for various optimizers such as stochastic gradient descent, RMS Prop, ADA Grad. This model is set with batch size 64, activation function ReLu, kaiming initialization algorithm and learning rate 0.001 and it is trained for 100 epochs. The figure 4.8 shows the performance of model with same parameters with Xavier weight initialization algorithm.

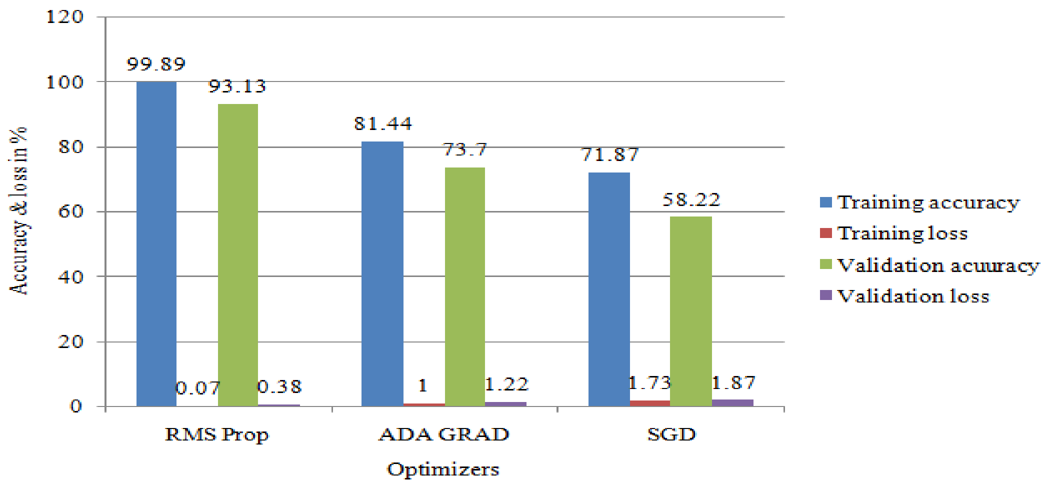

Figure 4.7 Performance of the model with learning rate 0.0001 and Kaiming weight initialization algorithm

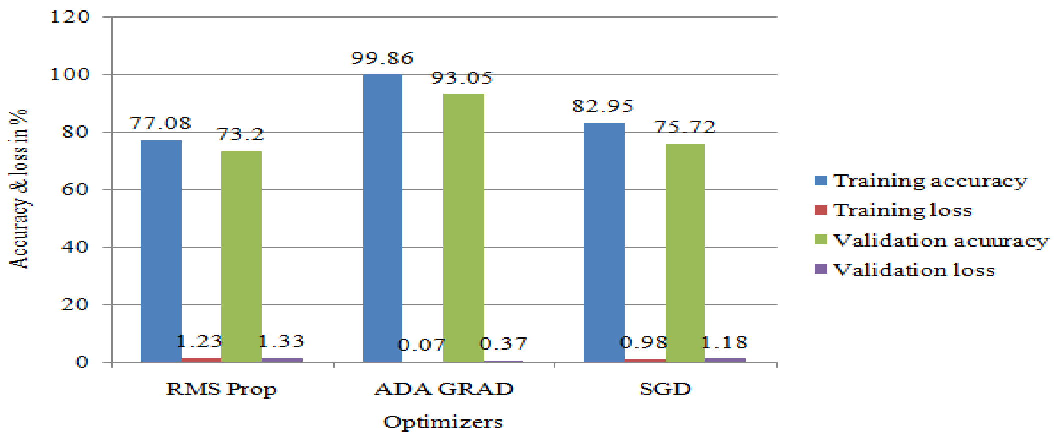

Figure 4.8. Performance of the model with learning rate 0.0001 and Xavier weight initialization algorithm

The model was initially trained with tamil-iso-character dataset, provided by hp-labs India. After hyper parameter tuning, the model had achieved 94.64% training accuracy and 92.14% validation accuracy. The handwritten Tamil paragraph is taken as an input image, and then the individual characters has been segmented and saved. The saved segmented Tamil character is given as an input to the model, achieved 85.15% accuracy. This is because there is lot of varieties in people's writing style. Even within the same paragraph, the handwriting of the same person differs. To improve the overall accuracy large amount of dataset is required.

5. CONCLUSION

An approach for offline handwritten Tamil text recognition model using CNN architecture is presented in this paper. Initially the individual character has been segmented from the paragraph, and then it is given to the trained model. The model is developed using convolutional neural network architecture and trained using hp-labs-Tamil-iso-character dataset. The model is fine tuned before using it for prediction. The insufficient precision is because of variety in writing pattern of individuals. For achieving better accuracy large amount of dataset with more writing styles is required. Hp labs provided Tamil character dataset for free, a special thanks to them which is very helpful for research in Tamil handwriting recognition model.

REFERENCES

[1] B.R. Kavitha, C. Srimathi, “Benchmarking on offline Handwritten Tamil Character Recognition using convolutional neural networks”, Journal of King Saud University - Computer and Information Sciences,2019,ISSN 1319-1578.

[2] M. Hanmandlu and O. V. R. Murthy, Fuzzy model based recognition of handwrittennumerals, Pattern Recognition 40(6) (2007) 1840–1854.

[3] D. Senapati, S. Rout and M. Nayak, "A novel approach to text line and word segmentation on odia printed documents," 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12), Coimbatore, India, 2012, pp. 1-6, doi: 10.1109/ICCCNT.2012.6396063.

[4] Vinotheni, S. Lakshmana pandian, G.Lakshmi, “Modified Convolutional Neural Network of Tamil Character Recognition”, Advances in Distributed Computing and Machine Learning Springer Singapore, DOI: 10.1007/978-981-15-4218-3_46, 2021.

[5] Shanthi, N.Duraiswamy, K, “A novel SVM-based handwritten Tamil character recognition system”, pattern analysis and application, springer London, doi: 10.1007/s10044-009-0147-0, vol.13, issue 2, pp.173-180, 2010.

[6] Boufenar, Chaouki&Kerboua, Adlen&Batouche, Mohamed., “Investigation on Deep Learning for Off-line Handwritten Arabic Character Recognition” Cognitive Systems Research. 50. 10.1016/j.cogsys.2017.11.002.

[7] Tiji M Jose , Amitabh Wahi, “ Recognition of Tamil Handwritten Characters using Daubechies Wavelet Transforms and Feed-Forward Backpropagation Networ”. International Journal of Computer Applications 64(8):26-29, February 2013.

[8] N. Aleskerova and A. Zhuravlev, "Handwritten Chinese Characters Recognition Using Two-Stage Hierarchical Convolutional Neural Network," 2020 17th International Conference on Frontiers in Handwriting Recognition (ICFHR), Dortmund, Germany, 2020, pp. 343-348, doi: 10.1109/ICFHR2020.2020.00069.

[9] Guha, Nibaran Das, Mahantapas Kundu, MitaNasipuri and K. C. Santosh, “An Efficient CNN Architecture for Handwritten Devanagari Character Recognition” International Journal of Pattern Recognition and Artificial Intelligence, 2020, Vol. 34, No. 12, 2052009.

[10]Chen, Yiqing, Comment on the work of Zhang et al. (2017, Journal of Inequalities and Applications). Journal of Inequalities and Applications, 2019(1), 186-192, doi:10.1186/s13660-019-2142-3.

[11]Bor, VojskeJugoslavije, “Text Line Segmentation With Water Flow Algorithm Based on Power FunctionDarkoBrodic 11” University of Belgrade, journal of electrical engineering VOL. 66, NO. 3, 2015, 132–141, DOI: 10.2478/jee-2015-0021.

[12]Razak, Z.—Zulkiflee,K.et al, “Off-LineHandwritingTextline Segmentation”, IJCSNS International Journal of Computer Science and Network Security, VOL.8 No.7, July 2008.

[13]Likforman-Sulem, Laurence, Zahour, Abderrazak, Taconet, Bruno, “Text line segmentation of historical documents” a survey International Journal of Document Analysis and Recognition (IJDAR)

[14]M. Karthi, R. Priscilla, K. Syed Jafer, “A Novel Content Detection Approach for Handwritten English letters” Procedia Computer Science, Volume 172,2020, Pages 1016-1025,ISSN 1877-0509.

[15]Ch. N. Manisha ,E. Sreenivasa Reddy and Y. K. Sundara Krishna “Role of Offline Handwritten Character Recognition System in Various Applications” International Journal of Computer Applications 135(2):30- 33, February 2016.

[16]Alfons Juan, Veronica Romero, Joan Andreu Sanchez, Nicolas Serrano, Alejandro H. Toselli and Enrique Vidal. “Handwritten Text Recognition for Ancient Documents” Workshop on Applications of Pattern Analysis. JMLR: Workshop and Conference Proceedings 11 (2010) 58-65, 2010.

[17]Maheswaran, S., Vivek, B., Sivaranjani. P., Sathesh, S., and Vignesh, K P., "Development of Machine Learning Based Grain Classification and Sorting with Machine Vision Approach for Eco-Friendly Environment," Journal of Green Engineering, vol. 10, no. 3, pp. 526-543, 2020.

[18]Sathesh, S., Pradheep, V A., Maheswaran, S., Premkumar, P., Gokul Nathan, S., and Sriram, P., “Computer Vision Based Real Time Tracking System to Identify Overtaking Vehicles for Safety Precaution Using Single Board Computer.” Journal of Advanced Research in Dynamical and Control Systems, vol. 12, no. SP7, July 2020, pp. 1551–61. *DOI.org (Crossref)*, doi:10.5373/JARDCS/V12SP7/20202258.

[19]4)V.R. Balaji, Maheswaran S, M. Rajesh Babu, M. Kowsigan, Prabhu E., Venkatachalam K,Combining statistical models using modified spectral subtraction method for embedded system,Microprocessors and Microsystems, Volume 73,2020.

[20]5)Malar, A.C.J., Kowsigan, M., Krishnamoorthy, N. S. Karthick, E. Prabhu & K. Venkatachalam (2020). Multi constraints applied energy efficient routing technique based on ant colony optimization used for disaster resilient location detection in mobile ad-hoc network. Journal of Ambient Intelligence and Humanized Computing, 01767-

[21]6)Miodrag Zivkovic, Nebojsa Bacanin, K. Venkatachalam, Anand Nayyar, Aleksandar Djordjevic, Ivana Strumberger, Fadi Al-Turjman, COVID-19 cases prediction by using hybrid machine learning and beetle antennae search approach,Sustainable Cities and Society, Volume 66,2021.

[22]7)Amin Salih Mohammed, Saravana Balaji B, Saleem Basha M S, Asha P N, Venkatachalam K(2020),FCO — Fuzzy constraints applied Cluster Optimization technique for Wireless AdHoc Networks,Computer Communications, Volume 154,Pages 501-508.