

Online Kannada Handwritten Characters and Numerical Recognition using CNN Classifier

Gandhana M H

P G Scholar

Department of Electronics and Communication
University B.D.T College of Engineering
Davanagere-577004, Karnataka, India

Dr. Lakshman Naik

Associate Professor,

Department of Electronics and Communication
University B.D.T College of Engineering
Davanagere-577004, Karnataka, India

Abstract— The objective of this paper is to recognize Kannada handwritten characters and numerals in a real-time application using CNN classifier. The data acquisition technique entails the capturing of data using a Graphics tablet that has 5080LPI resolution and an XP stylus pen. The sensor take-up the pointing and pen-up, pen-down and bobbing movement of the pen. Kannada manuscript includes 52 syllabary characters, these divided into 14 Vowels, 36 Consonants, 2 Special characters also 10 Kannada numerals 3100(62x50) dataset are used. Convolutional Neural Network(CNN) classifier used for character recognition purpose, training and testing of dataset carried out by CNN model. This proposed model provides 97.09% of testing accuracy and 0.212% as less testing error.

Keywords— Convolutional Neural Network(CNN), Character Recognition, Deep learning, Online handwritten recognition, Max-pool layer, Fully connected layer.

I. INTRODUCTION

In India, the Karnataka region people spoke Kannada which is the main Dravidian language. Currently, Deep Learning models have been more effective, it is implemented by the help of deep networks nothing but neural networks with multiple hidden layers. Online handwritten recognition is done by sensing and recording the pen pointing, pen-up and pen-down movements of the pen. This pen provides information by pressing the nib on the tablet within the writing window, this information is considered as input characters. For better accuracy and less error, our handwritten characters are used. Preprocessing technique is done to convert input images into binary data. Dataset split into 80% as training dataset and 20% as testing dataset. For character recognition Convolutional Neural Network(CNN) become streamer in the Deep Learning model, CNN improves the generalization ability of the model by controlling the over-fitting of the dataset.

II. LITERATURE REVIEW

In the field of character recognition, much research is done in different languages. In previous years Kannada based characters have many exceptional works in the field of Kannada character recognition. Character identification from an image is a method toward distinguishing and getting characters from the information image, Preprocessing operation performed to read, to convert image into binary form at the lowest level of abstraction[1]. The text data extract from scanned images to identify the Kannada letters and display or store for further usage, two methods are

proposed to recognize the handwritten Kannada characters that are Tesseract tool achieved 86% and Convolution Neural Network (CNN) archived 87% accuracy[2]. The character segmentation algorithm is proposed for Kannada handwriting recognition, segmentation results are validated using Support Vector Machine (SVM) classifier, Convolutional Neural Network (CNN) performs high-quality classification problem in the field of Computer Vision[3]. Online character recognition is a Real-time process, provides more accuracy than offline character recognition, the writing points are stored as a function of time in the online system[4]. Online recognition task proposed in which recognition takes place by recorded pen trace to reduce symbol recognition task that is offline[5]. Without any overlapping of characters obtained 96% accuracy for their own handwritten, CNN with appropriate configuration dataset accept the recognition rate for handwritten documents[6]. OHKC dataset provides 97.14% of recognition rate for online handwritten Kannada character recognition system[7].

III. DATASET

The dataset was collected in online mode using Graphic tablet which have 5080LPI resolution along with XP pen. The collected dataset contains our handwritten characters and numerical of Kannada. In this 52 syllabary characters used, these divided into 14 Vowels, 36 Consonants, 2 special characters also 10 Kannada numerical are considered and each character considers 50 comparable characters so totally 3100(62x50) dataset used. These datasets were collected by writing window which was displayed using python code size of this writing window considered as 255x255. The stylus pen traces the characters when the pen provide pressure on tablet the written characters as shown in Fig.1.

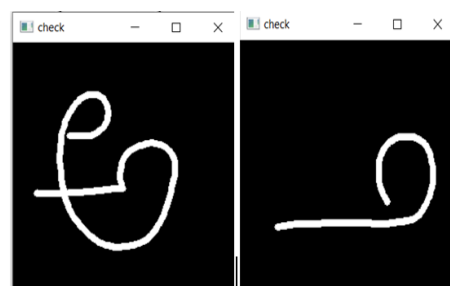


Fig.1 Kannada Character and Numerical

These can save using 's' button in the .jpg format, 'c' for clear and 'q' to quit the writing window, these images are used for further training and testing of the dataset.

IV. METHODOLOGY

Fig.2 describes the techniques entailed in this approached system.

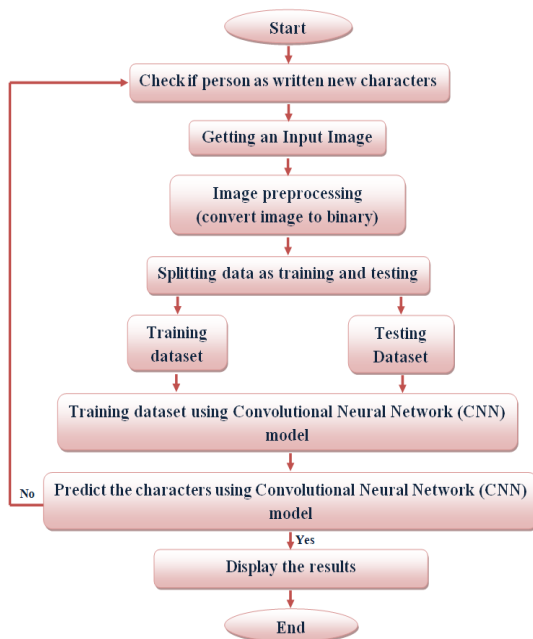


Fig.2 Flowchart for proposed work

A. Data Acquisition

The dataset is collected by virtue of a writing window, using Graphic tablet our handwritten characters are collected. The input data is accepted when the new characters are written in the writing window, subsequently, this new data have been regarded to be the further process.

B. Pre-Processing

In the procedure where the raw image is transformed into an appropriate processed image, the processed image will be used as an input for extracting different features. The new data is accumulated in a JPEG file format having image dimensions (255,255,3). The stored data can be recited and decode the data into RGB grids of pixels along with channels as shown in Fig.3, render input to Convolution Neural Network(CNN) by converting to floating point tensors. Rescale the pixel values ranges (0,255) to binary values that are [0, 1].

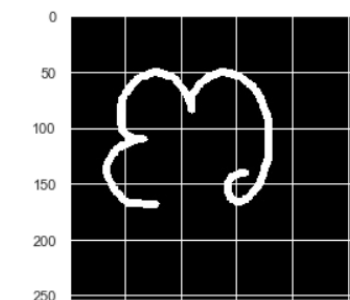


Fig.3 Preprocessing of data

C. Splitting The Data

The obtained data is split into 80% of training data and 20% of testing data, furthermore, training data cleaved into training images (2480,255,255,3) and training labels (620,255,255,3) comparably testing data cleaved into testing images (2480,62) and testing label (620,62).

D. Convolutional Neural Network(CNN)

Convolutional Neural Network(CNN) is a kind of deep neural network model specially used for supervised machine learning. CNN composed of input, hidden and output layer, the hidden layer consist of convolutional layer, pooling layer and fully connected layer shown in Fig.4.

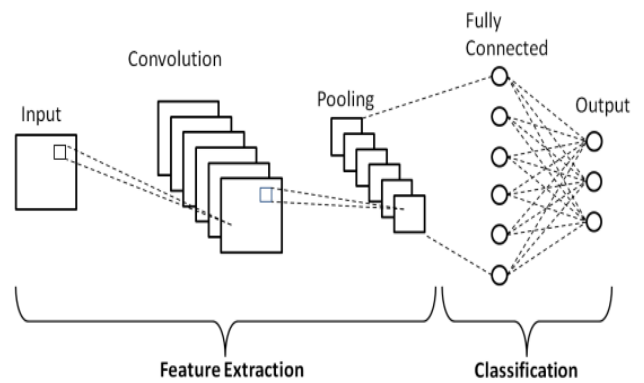


Fig.4 Convolutional Neural Network

Usually, the image is taken as input from the input layer, CNN compares the piece of the image by piece and called as features, so by matching the similar features from different images available in the dataset. The convolutional layer convolves input by extracting high-level features using feature detectors and proceed its consequence to the next layer, which is similar to the retaliation of neurons in the visual cortex to a specific stimulus. The ReLU layer helps with making the model non-linear. In Max-pooling the maximum values are taken from the window, which reduces dimension, computation and over-fitting by integrating the output of neuron clusters at one layer into a single neuron of the next layer. A fully connected layer connects every neuron in a single layer to every neuron of the next layer. In the output layer, the number of neurons is equivalent to several classes. Finally, the input data and output data are displayed in the designed canvas.

V. EXPERIMENTAL RESULTS

In this proposed model, the dominant python(3.6) library tools are used as propounded in Table.1 The CNN model is a Sequential type model, as shown in Fig.5 in the input layer the size of the input is (255,255,3) and kernel size is (3,3) is considered and three hidden layers are used.

TABLE I. PYTHON LIBRARY TOOLS

Library tool	Version	Description
Tensorflow	1.15.2	Implement Deep Neural Network
Keras	2.3.1	Beneath for Tensorflow
tqdm	2.2.4	Show the progress bar
opencv	4.1.1.26	Develop Real-time computer vision
matplotlib	2.23	Numerical mathematical extension for numpy
seaborn	11.1	High-level interface for drawing attractive and informative statistical graphics
Scikit-learn	0.24.2	Provides many unsupervised & supervised learning algorithms

```
# Model Details
print(model.summary())
Model: "sequential_1"

Layer (type)                Output Shape                Param #
-----
conv2d_1 (Conv2D)           (None, 253, 253, 32)        896
max_pooling2d_1 (MaxPooling2 (None, 126, 126, 32)        0
conv2d_2 (Conv2D)           (None, 124, 124, 64)        18496
max_pooling2d_2 (MaxPooling2 (None, 62, 62, 64)        0
conv2d_3 (Conv2D)           (None, 60, 60, 64)         36928
max_pooling2d_3 (MaxPooling2 (None, 30, 30, 64)        0
conv2d_4 (Conv2D)           (None, 28, 28, 128)        73856
max_pooling2d_4 (MaxPooling2 (None, 27, 27, 128)        0
dropout_1 (Dropout)         (None, 27, 27, 128)        0
flatten_1 (Flatten)         (None, 93312)              0
dense_1 (Dense)             (None, 124)                11570812
dropout_2 (Dropout)         (None, 124)                0
dense_2 (Dense)             (None, 62)                 7750
-----
Total params: 11,708,738
Trainable params: 11,708,738
Non-trainable params: 0
```

Fig.5 Configuration of CNN model in python

Totally 11708738 parameters obtained in this all 11708738 parameters are trained efficiently.

TABLE II. CONFIGURATION OF CNN FOR TRAINING

Epochs	Training Accuracy	Training Loss
Epoch 1/20	66.29%	1.43%
Epoch 4/20	97.74%	0.082%
Epoch 12/20	98.91%	0.042%
Epoch 19/20	99.07%	0.026%
Epoch 20/20	99.23%	0.025%

CNN model dealing with 2-Dimensional images hence convolution process and max-pooling operations in 2-Dimension only. To compile CNN Adam optimizer is used. Table.2 mentions the accuracy and loss values of different epochs, using the model. fit function CNN model has trained 20 epochs.



Fig.6 Training and Validation Accuracy

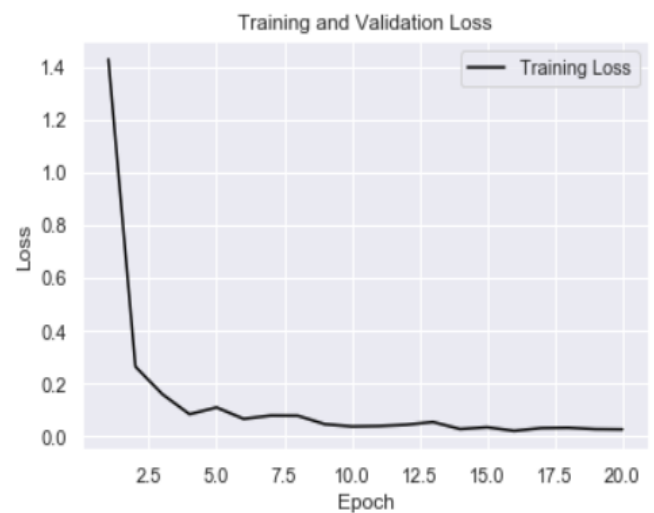


Fig.7 Training and Validation Loss

The better accuracy is obtained as the number of iterations is raised, once the model is trained and get higher accuracy then that model is preferred for testing. Fig.6 represents the training and validation accuracy and Fig.7 loss represents the training and validation loss.

Accuracy and Loss percentages of Training and Testing data:

- Training Accuracy: 99.23%
- Training Loss: 0.0251%
- Testing Accuracy: 97.09%
- Testing Loss: 0.210%

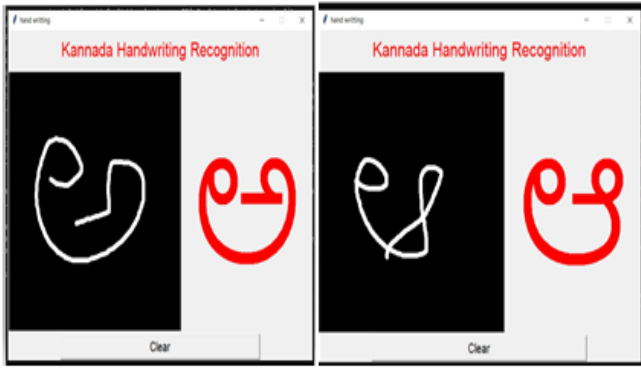


Fig.8 Recognized Kannada Characters

Using thinter and python code we draw buttons, input boxes, in black canvas the user can write characters and in white canvas the recognized labeled data will be displayed.

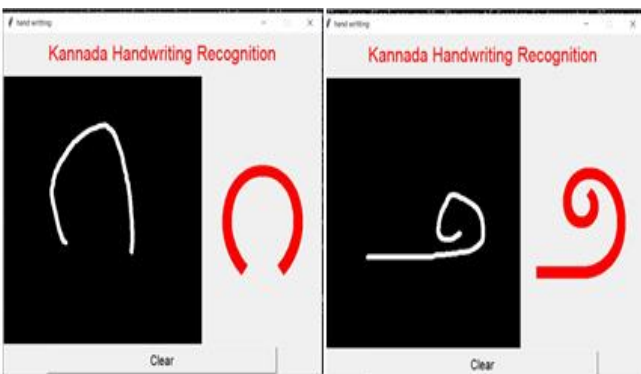


Fig.9 Recognized Kannada Numericals

Fig.8 represents Kannada characters and Fig.9 represents Kannada numerals. The clear option is used to clear the data which is written in user canvas.

VI. CONCLUSION AND FUTURE WORK

In this proposed work, an online Kannada handwritten characters and numerical recognition is concluded using Convolutional Neural Network(CNN) model. We have obtained 97.01% of accuracy and 0.212% of less testing error, for our handwritten characters. CNN detects the important features spontaneously without any human supervision. The performance of the classifier can enhance to get higher accuracy and less loss for a large number of the dataset. This paper provides better knowledge for further implementation of Kannada handwritten character recognition in real-time.

REFERENCES

- [1] Dr. Manish Kaushik, "Hand-written Character Identification from an Image by using Digital Image Processing", Subodh Journal of Recent Trends in Information Technology, volume 10, Issue-02, June 2019.
- [2] Roshan Fernandes and Anisha P Rodrigues, "Kannada Handwritten Script Recognition using Machine Learning Techniques", IEEE,2019.
- [3] Ramesh G, Sandeep Kumar N, Champa H N, "Recognition of Kannada Handwritten Words using SVM Classifier with Convolutional Neural Network ", IEEE, 2020.
- [4] Anisha Priya, Surbhi Mishra, Saloni Raj, Sudarshan Mandal and Sujoy Datta, "Online and Offline Character Recognition: A Survey", International Conference on Communication and Signal Processing, April 6-8, 2016.
- [5] Sergey Arseev, Leonid Mestetsky, "Handwritten Text Recognition Using Reconstructed Pen Trace with Medial Representation", IEEE, 2020.
- [6] Asha K, Krishnappa H K, "Kannada Handwritten Document Recognition using Convolutional Neural Network", IEEE International Conference on Computational Systems and Information Technology for Sustainable Solutions, 2018.
- [7] Rajani Kumari Sah, DR. K Indira, "Online Kannada Character Recognition using SVM Classifier", IEEE, 2017.
- [8] Qisheng Hu, "Evaluation of Deep Learning Models for Kannada Handwritten Digit Recognition", International Conference on Computing and Data Science, 2020.
- [9] Mamatha H R, Srikanta Murthy K, Veeksha A V, Priyanka S Vokuda and Lakshmi M, "Recognition of Handwritten Kannada Numericals using Directional Features and K-Means", International Conference on Computational Intelligence and Communication System, 2011.
- [10] Chaitra D, Dr. K Indira, "Handwritten Online Character Recognition for Single Stroke Kannada Characters", IEEE International Conference on Recent Trends in Electronics Information & Communication Technology(RTETCT) May 19-20, 2017.