IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

Kannada MNIST Digit Recognition Using CNN and RNN Model

Akshitha L S

Department of Computer Science and Engineering,

University Visvesvaraya College of Engineering, Bangalore, India-560001

Abstract: Recognition of handwritten digits is a interesting research topic in Optical Character Recognition (OCR) in recent years. In this work, a CRNN model is proposed which is the combination of convolutional neural network (CNN) and Recurrent neural network (RNN). In this model, GRU is used to replace the CNN fully connected layer part to achieve high recognition accuracy Firstly, CNN ensures that the best features of the original image can be extracted using multiple convolutional and pooling layers. Secondly, GRU is a fast recognition method, which established a sequential relationship between features in the hidden layer. Experiment result performed on Kannada MNIST handwritten digit dataset achieves 98.12% training accuracy and 98.95% validation accuracy and also achieved higher precision, recall and F1 score ratio.

Keywords: CNN, RNN, GRU and Kannada MNIST

I. INTRODUCTION

Deep learning & machine learning adopts an important function in computer technology & artificial intelligence. With the deployment of machine learning, human exertion can be moderated in recognizing, learning, predicting & lot more regions. It is a wild developing area of computer discipline that builds on its way to all other domains. Substantial space in the region which is well-organized & all-purpose handwritten digit recognition. The handwritten digit recognition has several possible real applications such as marks digitization, banking utilities, reading postal code & tax form. There has been an enormous advancement in the area of machine learning with the application of computer vision in it. Disparate traditional methods which comprise different pre-processing steps, deep learning automatically identifies the features. Consequently, deep learning typically depends on the data & hence can be applied to resolve different types of problems, including handwritten characters & numerals recognition. The handwriting recognition procedure can be fragmented down into two phases: First the feature extraction phase & Second the classification phase. The significant role in Feature extraction is getting high accuracy rates. However, along with the appropriate pre-processing of data also gives to high accuracy. Several research activities are made in the esteem for English numerals & imposing outputs are obtained. However, there is room for more development when it comes to Kannada numerals.

Deep learning is said to be wild developing & technically developing region of data science that is creating its particular path into the all other domains. A significant experiment in the region of deep learning that is towards the progress an effective & all-purpose handwriting classification technology that meets the attention of the wide variety of the applications. Transformation of the diverse scripts, legislative bodies, banks, offices, reading sign boards, applications of archaeology, support to blind, areas of literature are some of the significant zones of all the application of handwriting recognition task. Generally digitisations of enormous number of the documents that can be effortlessly carry out with all the benefits of handwriting recognition equipment. One such application is recognizing the scripts of several languages.

In the earlier few years, the Convolutional Neural Network (CNN) model consumes enthusiastically engaged for handwritten digit recognition from the standard MNIST database. CNN is unique categories of the neural networks that are generally used to investigate images. It proceeds images as input & extracts diverse features from the image. It comprises the input layer, many hidden layers & finally the output layer. CNNs are very beneficial & achievable in allocating with patterns involving spatial arrangements, thus useful in recognition of handwritten characters & digits. The main benefits of the CNN when associated to the traditional machine learning techniques are that it subsequently discriminates the important features in the image with no human oversight. A CNN incorporates the feature extraction phase & classification phases & involves negligible pre-processing & the extraction of feature determinations. Convolutional Neural Network can extract prosperous & consistent features that are automatically from the images. Furthermore, CNN can deliver substantial accuracy of the recognition uniform though if that is the only a slight training datas that are presented. Recurrent neural network (RNN) is an influential domestic of connectionist models which detention period dynamic forces via cycles in the graph. Distinct feed forward neural networks, recurrent networks that can development examples each at a time, that retaining a state or memory, that reproduces a subjectively

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

long context window. Whereas these networks ingests difficult to train & frequently contain number of parameters, optimization techniques, recently improvements in the network architecture & similar computation that have allowed large scale learning with recurrent networks. Earlier few years, the systems are established on state of the art. Long-Short Term Memory & bidirectional Recurrent neural network architecture have established record setting presentation on tasks as diverse as language translation, image captioning & handwriting recognition. The stimulating part of Indian script handwritten classification is the division among the similar mechanism. Occasionally an actual minor part is the unique mark between two numerals. These minor unique portions increase the complexity of the recognition & reduce recognition accuracy. Since of the writing styles of diverse individuals, similar numerals may proceeds diverse shapes & equally one or more dissimilar number of a scripts that may consume similar shape. These kinds of features increase the complexity of recognition technique. The MNIST dataset is a typical dataset used universal consisting of 60,000 dataset samples for the training the dataset & 10,000 dataset samples for the testing dataset. Kannada language is the authorized language of the state of Karnataka, India, vocalized by over 50 million folks all over the world. Kannada MNIST delivers a standard style dataset for numbers in kannada. Fig. 1.1 signifies the digits from 0 - 9 in Kannada.

Fig 1.1: Kannada digits

Conversely, there is slight effort on any datasets numbers in the language. These are frequent problems in supplementary research for neural networks resolutions, emerging NLP & algorithms natively for the digits of Kannada & there is an instantaneous essential for the datasets & additional implements in order to additional broadcast research parts in Kannada dataset. Similar to the MNIST dataset, the Kannada-MNIST dataset was presented which comprehends the Kannada-MNIST dataset as well as Dig-MNIST dataset. The Dig-MNIST is a very stimulating dataset when associated to the Kannada-MNIST as the dig-MNIST was produced with the support of helpers that were non-native workers of kannada language. The Handwritten digit recognition is usually distributed into two phases: (a) feature extraction & (b) classification. These phases that could be apprehends by the Convolutional Neural Network. Conversely, subsequently the Convolutional Neural Network has an outstanding presentation in feature extraction step; numerous hybrid structures such as Convolutional Neural Network & SVM, Convolutional Neural Network & HMM (Hidden Markov Model), & Convolutional Neural Network & RNN (Recurrent Neural Network) have been recommended to recover the recognition accuracy. In this work, it was founded that a Convolutional Neural Network structure, which merges a structure of Convolutional Neural Network & GRU is suggested to additional progress reducing running time & recognition accuracy. The Recurring network GRU is a distinctive Recurrent Neural Networks model that had been functional extensively in arrangement difficulties. The inspiration of this effort has two main succeeding features: Initially, the arrangement preserves the CNN presentation in the feature extraction process to extract the features from the original images. Furthermore, the GRU is able to change handwritten digit classification into recognition problematic to competently accomplish the classification task.

II. LITERATURE REVIEW

Baoguang Shi. et Al., [1] projected technique obtain the neural network architecture, termed Convolutional Recurrent Neural Network (CRNN), that is assimilates the compensations of the Convolution Neural Network & Recurrent Neural Networks. Convolutional Recurrent Neural Network is capable to the precede input images of the variable dimensions & produces expectations with diverse length. It straight turns on bristly level of the labels needful no detailed explanations for each individual component in the training stage. Furthermore, as Convolutional Recurrent Neural Network leaves fully-connected layers used in conventional neural network, it consequences in additional dense & efficient models. The Convolutional Recurrent Neural Network accuracies of Real-World are 84.0%/0.30. Youssouf C. et Al., [2] suggested the structure for the feature dataset estimation established upon collective settings. Using a biased vote grouping of Recurrent neural networks classifier, all the data were train with a specific feature dataset. The grouping was demonstrated in the framework as the combination model & two approaches for weights approximation are labeled. The foremost involvement of the paper is to enumerate the significance of feature set throughout the combination weights that reflects the complementarity strength. The author selected the RNN classifiers since it is a state of the art performance. Also, deliver the initial feature set for the classifier. Evaluates the number of feature gatherings on the IFN / ENIT & RIMES datasets of & Latin & Arabic script correspondingly. The subsequent amalgamation model is modest with state of the systems. Sajedi et al., [3] suggests a regularization of research mechanism on OCR in Persian language. A database termed Persian Handwritten Optical Digits & Numbers are used for classification determination. The projected technique uses Support Vector Machines (SVM) with RBF & K-Nearest

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

Neighbour (KNN), linear kernals & Polynomial kernels are working in SVM for calculating the efficiency of the removed features. The planned method is said to accomplish a greater recognition rate up to 99%.
Mirabdollah et al., [4] had been estimated the additional features of the DAG vectors, where the images are distributed into windows & blocks of typical sizes, & the typical of values of all pixels in each windows are attached to the descriptors to acquire the feature vectors. Karthik et al., [5] has been gathered all the vowels & consonants distinctly & used more than 400 sample images per character to trains the model of the deep belief networks. The determination is supportive in considerate various approaches, which is used for resolving the modeled problematic of handwritten Kannada letter recognition. They have demanded an average accuracy of 97.04 %. P. Bannigidad et al., [6] have been prepared the determinations towards the renovation of dishonored Kannada handwritten manuscripts or hastapratis by together global binarization methods & special local methods, by the removal of uneven background enlightenment. Yadav et al., [7] has focused on the exertion towards the field. They designate a method to accomplishment recognition on Kannada characters of handwritten datasets or the contemporary in the images, using the Histogram Oriented Gradients (HOG) descriptors, for extraction of the feature from the images of the offline handwritten kannada characters, & recollecting a deep learning ideal (Neural networks and SVM) for concluding classification. This kind of determination demonstrated the evaluation of recognition of the accuracy among the two classifiers.
Chherawala et al. [8] had been stated the word images are used to extract the features & the handwritten script is predictabled from the features. They established model as a presentation of the recurrent neural networks. The Recurrent neural networks classifier has been used as a weighted vote grouping, wherever the consequence of feature datasets is documented by the weights & groupings. Jaderberg et al., [9] presented the final system which is used for the localizing; text spotting & recognizing the text in natural scenes in images & text based images are retrieval. The system is established on a region suggestion instrument for detecting the deep convolutional neural networks for recognition. The Pipeline uses an arrangement of balancing suggestion combination methods to confirm great recall, & a wild succeeding filtering phase for enlightening precision. Analyze the phases of pipeline, demonstrates state of the art presentation Kannada Digit Recognition Using CNN & RNN Model throughout. Accomplish several conducting experimentations across a number of standard end to end text spotting & text based images are retrieval by the datasets, which present a large growth in overall preceding approaches. Finally, establish a real-life application of the text spotting of the system to permit number of hours of news film to be quickly searchable via a text request. Roy et al. in [10] focused a method for the motions following which trusted on the videos that provender from all-purpose device such as camera, thus creating the airwriting framework that are available for the common determinations. The shortage of the delimiters among characters in the airwriting medium positions an experiment conducted for recognition of characters in a nonstop stream.

III. PROBLEM STATEMENT AND OBJECTIVES

This work mainly focuses on handwritten Kannada digits recognition, by using the state-of-the-art classifier, which gives very high accuracy. This effort is focused on using CNN with RNN on the Kannada-MNIST dataset to recognize the digits efficiently and overcome the limitations of the existing works. This work likewise analyses the limitations in different methods that are being used as the solution for the issue. This allows the models to effectively learn the different possible variations they may encounter and classify each correctly.

The goal of this work is to solve Handwritten Kannada Digits recognition, which is a very challenging topic in recent years. This lets the computer to understand Kannada digits that is written manually by the user using CNN and RNN.

The objectives of this work are:
- To achieve better accuracy in training and validation.
- To maintain higher precision, recall and F1score ratio.
- To reduce the loss in training and validation.
- To reduce the number of epochs.

IV. SYSTEM ARCHITECTURE

The CRNN model is a combination of Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN). The Proposed method of System Architecture is illustrated in the Fig 4.1. CNN is a multi-layer neural network structure which is composed of multiple convolutional layers, Relu layers, pooling layers and fully connected layers, in which the convolutional layers and pooling layers are used to extract features of the handwritten digits. Recurrent Neural Networks are capable to capture in its hidden state temporal and spatial features within a sequence of inputs, so they are suitable for the handwriting recognition. Gated recurrent units (GRUs) are a gating mechanism in recurrent neural networks. The GRU networks replace CNN fully connected layers to transform the classification task into a sequence

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

task, in which the classification results of each feature map are added to the next feature map classification calculation in the same hidden layer to improve the CNN recognition performance.

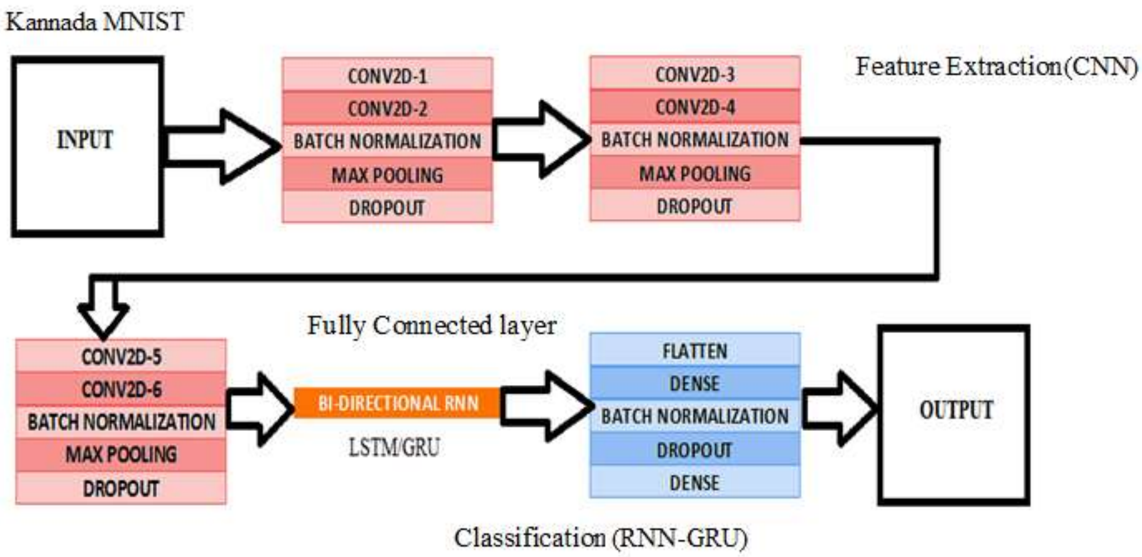

Fig 4.1: Proposed method of System Architecture

V. PROPOSED METHOD

This work uses the combination of Convolutional Neural Network and Recurrent Neural Network to solve the given problem statement. Convolutional Neural Network is used to regularize the features extraction and Recurrent Neural Network is used for classification of Kannada MNIST digit recogntion.

A. Kannada MNIST dataset

Fig 5.1: Sample Kannada MNIST dataset

Kannada is the official and the most spoken language in Karnataka state of India. It is one of the oldest Dravidian languages of India just like Tamil. The earliest inscription having all 9 Kannada numerals have been engraved in the Gudnapur Inscription which dates back to the time of Kadamba Ravivarma (485 A.D. to 519 A.D.). The symbols used to represent 0 to 9 in kannada as shown in Fig. 6.1 are distinct from the modern Hindu-Arabic numerals. The Kannada MNIST dataset has a training set which comprises of 60,000 images converted to grayscale, each of the size 28 x 28 and a testing set which comprises of 5,000 images converted to grayscale with the same size and are divided equally among 10 classes.

B. Pre-processing

Pre-processing is the method to convert the data image to a picture that is better suited for extracting features. As pre-processing methods slant correction, binarization, skew detection and noise reduction and morphological operations are used.

The methods for pre-processing are explained below.

- **Skew detection of images and correction:** Fourier transform is used to perform this operation.

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

- **Binarization:** This process is basically used to convert all the gray scale images into binary images through a method of global Otsu threshold approach. Otsu's thresholding method involves iterating through all the possible threshold values and calculating a measure of spread for the pixel levels each side of the threshold, i.e. the pixels that either falls in foreground or background. The aim is to find the threshold value where the sum of foreground and background spreads is at its minimum.
- **Noise removal:** This step is performed to remove the noise with the technique called Median filtering. This technique reduces maximum background noise from the image.
- **Normalization:** The method of transforming the images that are of randomsized image to standard-sized image is called normalization. This step is used to eliminate variation between characters between groups. Before the process of normalization, all the extra white spaces in the image are removed. Lastly, the given input image is normalized to a standard 28x28 resolution.
- **Thinning:** This operation of thinning is performed by removing the binary-valued image regions to lines approximating the region's skeletons to make the image crisper. The images that are preprocessed are ready for use in further phases of extraction and classification of features.

C. Feature extraction using Convolutional Neural Networks

The features are extracted from the input images using convolutional neural network layers (CNNs). CNN is a multi-layer neural network structure which is composed of multiple convolutional layers, pooling layers and fully connected layers, in which the convolutional layers and pooling layers are used to extract features of the handwritten digit character. In addition, the fully connected layers are used to classify the handwritten digit from the extracted feature maps. Each layer input is connected to the previous layer output, and its output passes to the next layer.

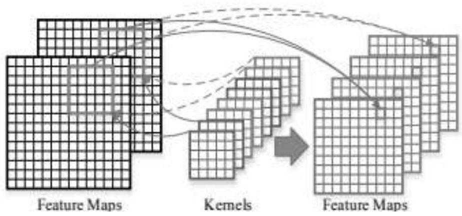

Fig 5.2: Operation on convolutional layer

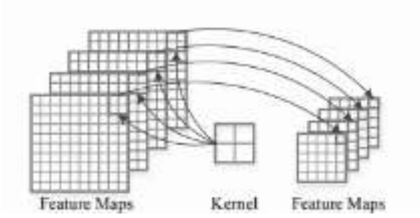

Fig 5.3: Operation on pooling layer

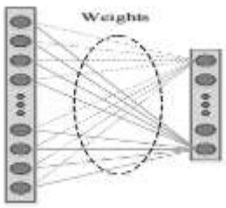

Fig 5.4: Operation on fully connected layer

In the convolutional layer as shown in Fig 5.2, the output feature maps are computed via sliding 3 by 3 kernels on the previous layer feature maps. In the pooling layer as shown in Fig 5.3, the number of output maps is unchanged, and each map size reduces from the input feature size by using a kernel size of 2 by 2 and a polling stride of 2. The number neurons of the first fully connected layer are obtained by converting all features of the last pooling layer into one vector. In the fully connected layer as shown in Fig 5.4, there is unidirectional full connection from the previous layer to current fully connected layer, and no connection between neurons in the same layer. Each neuron value is computed by Equation 1, where w is the weight of the connection from l-1 layer to l layer, blt is a bias, and f is an activation function

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

$h^l_t = f(\sum w.h^{l-1}_t+b^l_t)$ (1)

Sometimes batch normalization or instance normalization layer is used after CNN layer. The main purpose of this kind of layers is to standardize our input features to each have a zero mean and variance of one. Both types of normalization are quite similar, except for the number of input tensors that are normalized together.

D Classification using Recurrent Neural Networks

In CRNN, fully-connected layers at the end of CNN are not used; instead, the output from the convolutional layers (feature maps) is transformed into a sequence of feature vectors. These vectors are then fed into some type of bidirectional RNN (GRU in our case). GRU is a concise version of the long short term memory model (LSTM).

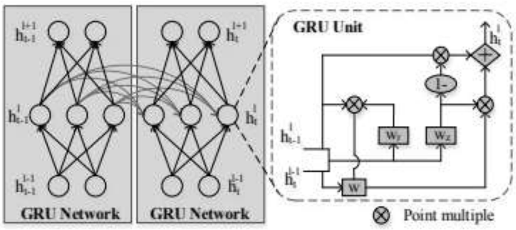

Fig 5.5: Architecture of gate recurrent unit network.

The LSTM has three gates including input gate, forget gate and update gate. Meanwhile, the GRU is simpler than the LSTM since it has only two gates, which are reset gate and update gate. These gates are used to determine whether the information is useful or not. The useful information is reserved while useless information is forgotten. As shown in Fig 5.5, in GRU structure, Wz, Wr and W represent the update gate, reset gate and candidate information, respectively.

In addition to fully connect with the neurons in the previous hidden layer at time t, each neuron in the GRU hidden layer is fully connected to all neurons in the current hidden layer at time t-1. The l^{th} hidden layer output is computed by the following equation:

$h^l_t = (l-z_t)*h^l_{t-1}+z_t*h^{-l}_t$ (2)

where z_t is an update gate, and h^{-l}_t is the candidate memory information. Moreover, z_t is given by equation 3 controls how much information of the previous memory and the current memory will be forgotten or to be added.

$z_t = \text{sigmoid}(Wz[h^l_{t-1},h^{l-1}_t])$ (3)

The candidate information value h^{-l}_t is calculated by Equation 4:

$h^{-l}_t = \tanh(W.[r_t*h^l_{t-1},h^{l-1}_t])$ (4)

where r_t defined by Equation 5 is the GRU reset gate which efficiently resets the information in the memory.

$r_t = \text{sigmoid}(W_r.[h^l_{t-1},h^{l-1}_t])$ (5)

Sigmoid and tanh are activate functions, expressed as:

$\text{sigmoid}(x) = \frac{1}{1+e^{-x}}$ (6)

$\tanh(x) = \frac{e^x-e^{-x}}{e^x+e^{-x}}$ (7)

In comparison with the CNN fully connected layers, the calculated information by the GRUs model contains more the historical state, in which the neuron value at time t is not only determined by the data in the previous layer at time t, but also is determined by the data stored in the GRU cell at time t-1 (as Equation 2). Since in the same CNN fully connected layer, neural units are not connected to each other. The GRU networks replace CNN fully connected layers to transform the classification task into a sequence task, in which the classification results of each feature map are added to the next feature map classification calculation in the same hidden layer to improve the CNN recognition performance. The proposed CRNN architecture is shown in Fig 4.1 and layers of CRNN proposed model is as shown in Fig 5.6. Furthermore, the dropout method which can randomly delete some neural units in the hidden layer is used to avoid the possible over-fitting problem in the CRNN model.

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 28, 28, 64)	640
conv2d_1 (Conv2D)	(None, 28, 28, 64)	36928
batch_normalization (BatchNo	(None, 28, 28, 64)	256
max_pooling2d (MaxPooling2D)	(None, 14, 14, 64)	0
dropout (Dropout)	(None, 14, 14, 64)	0
conv2d_2 (Conv2D)	(None, 14, 14, 128)	73856
conv2d_3 (Conv2D)	(None, 14, 14, 128)	147584
batch_normalization_1 (Batch	(None, 14, 14, 128)	512
max_pooling2d_1 (MaxPooling2	(None, 7, 7, 128)	0
dropout_1 (Dropout)	(None, 7, 7, 128)	0
conv2d_4 (Conv2D)	(None, 7, 7, 256)	295168
conv2d_5 (Conv2D)	(None, 7, 7, 256)	590080
batch_normalization_2 (Batch	(None, 7, 7, 256)	1024
max_pooling2d_2 (MaxPooling2	(None, 7, 3, 256)	0
dropout_2 (Dropout)	(None, 7, 3, 256)	0
time_distributed (TimeDistri	(None, 7, 768)	0
bidirectional (Bidirectional	(None, 7, 64)	153984
flatten_1 (Flatten)	(None, 448)	0
dense (Dense)	(None, 256)	114944
batch_normalization_3 (Batch	(None, 256)	1024
dropout_3 (Dropout)	(None, 256)	0
dense_1 (Dense)	(None, 10)	2570

Total params: 1,418,570
Trainable params: 1,417,162
Non-trainable params: 1,408

Fig 5.6: Layers of proposed CRNN model

VI. RESULTS AND ANALYSIS

The Kannada MNIST dataset contains 60,000 images belonging to 10 classes for training and 5,000 images for testing. The gray scale images of size 28 X 28 was the input to our model as displayed in Fig 6.1 and trained using the RMSprop optimizer having a learning rate equal to 0.00001. The batch size of 64 was used for training. A dropout of 0.25 and batch normalization was added after every two convolutional layers to avoid over fitting during training. The training is stopped after 10 epochs.

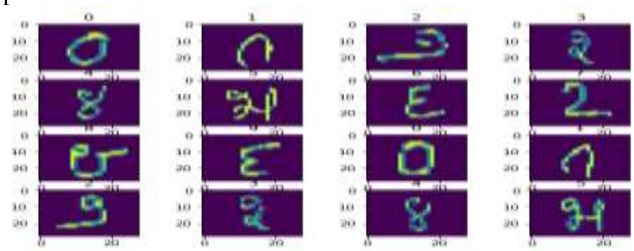

Fig 6.1: Plotted sample Kannada MNIST dataset

The network starts with the traditional 2D convolutional neural network followed by batch normalization, RELU activation, max-pooling and dropout with a dropout rate of 25%. Six convolution layers are placed in a sequential

IARJSET
ISSN (O) 2393-8021, ISSN (P) 2394-1588
International Advanced Research Journal in Science, Engineering and Technology
Vol. 8, Issue 11, November 2021
DOI: 10.17148/IARJSET.2021.81125

manner with their corresponding activations. The convolutional layers are followed by the permute and the reshape layer which is very necessary for CRNN as the shape of the feature vector differs from CNN to RNN. The convolutional layers are developed on 3-dimensional feature vectors, whereas the recurrent neural networks are developed on 2-dimensional feature vectors.The permute layers change the direction of the axes of the feature vectors, which is followed by the reshape layers, which convert the feature vector to a 2-dimensional feature vector. The RNN is compatible with the 2-dimensional feature vectors. The proposed network consists of bidirectional GRU layers with GRU cells in each layer where cell depends on the number of classes of the classification performed using the corresponding network. The bidirectional GRU (Gated recurrent unit) is used instead of the unidirectional RNN layers because the bidirectional layers take into account not only the future timestamps but also the future timestamp representations as well. Incorporating two-dimensional representations from both the timestamps allows incorporating the time dimensional features in a very optimal manner. Finally, the output of the bidirectional layers is fed to the time distributed dense layers followed by the Fully connected layer. The average training accuracy and validation accuracy are represented in Fig 6.2.

```
/usr/local/lib/python3.7/dist-packages/keras/engine/training.py:1915: UserWarning: `Model.fit_generator` is deprecated and will be removed
  warnings.warn('`Model.fit_generator` is deprecated and '
Epoch 1/10
843/843 - 1054s - loss: 0.2130 - accuracy: 0.9357 - val_loss: 0.0652 - val_accuracy: 0.9845
Epoch 2/10
843/843 - 1019s - loss: 0.0783 - accuracy: 0.9779 - val_loss: 0.0410 - val_accuracy: 0.9895
Epoch 3/10
843/843 - 1015s - loss: 0.0613 - accuracy: 0.9821 - val_loss: 0.0606 - val_accuracy: 0.9820
Epoch 4/10
843/843 - 1014s - loss: 0.0531 - accuracy: 0.9851 - val_loss: 0.0319 - val_accuracy: 0.9917
Epoch 5/10
843/843 - 1013s - loss: 0.0521 - accuracy: 0.9852 - val_loss: 0.0283 - val_accuracy: 0.9928
Epoch 6/10
843/843 - 1016s - loss: 0.0475 - accuracy: 0.9871 - val_loss: 0.0311 - val_accuracy: 0.9908
Epoch 7/10
843/843 - 1014s - loss: 0.0443 - accuracy: 0.9882 - val_loss: 0.0361 - val_accuracy: 0.9900
Epoch 8/10
843/843 - 1015s - loss: 0.0434 - accuracy: 0.9882 - val_loss: 0.0455 - val_accuracy: 0.9885

Epoch 00008: ReduceLROnPlateau reducing learning rate to 0.0024999999441206455.
Epoch 9/10
843/843 - 1014s - loss: 0.0311 - accuracy: 0.9914 - val_loss: 0.0370 - val_accuracy: 0.9910
Epoch 10/10
843/843 - 1019s - loss: 0.0304 - accuracy: 0.9917 - val_loss: 0.0215 - val_accuracy: 0.9952
```

Average Accuracy

```
[ ] np.mean(history.history['accuracy'])
    0.9812592625617981
```

Average Validation Accuracy

```
[ ] np.mean(history.history['val_accuracy'])
    0.9895999968051911
```

Fig 6.2: Average Training accuracy and Validation accuracy

The classification report visualizer displays the precision, recall, F1, and support scores for the model. The classification report of the proposed model is shown in Fig 6.3. A Classification report is used to measure the quality of predictions from a classification algorithm. How many predictions are True and how many are False. More specifically, True Positives, False Positives, True negatives and False Negatives are used to predict the metrics of a classification report.

	precision	recall	f1-score	support
0	1.00	0.99	1.00	557
1	0.99	1.00	1.00	577
2	1.00	1.00	1.00	626
3	1.00	0.99	0.99	592
4	1.00	0.99	0.99	627
5	0.99	1.00	1.00	573
6	0.99	0.99	0.99	576
7	0.98	1.00	0.99	627
8	1.00	1.00	1.00	634
9	1.00	0.99	1.00	611
accuracy			1.00	6000
macro avg	1.00	1.00	1.00	6000
weighted avg	1.00	1.00	1.00	6000

Fig 6.3: Classification Report

Confusion Matrix is a summary of prediction results on a classification problem. The number of correct and incorrect predictions are summarized with count values and broken down by each class. This is the key to the confusion matrix. The confusion matrix shows the ways in which your classification model is confused when it makes predictions. The Confusion Matrix of proposed model is represented in Fig 6.4. Classification accuracy alone can be misleading if you have an unequal number of observations in each class or if you have more than two classes in your dataset. Calculating

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

a confusion matrix can give you a better idea of what your classification model is getting right and what types of errors it is making. In Fig 6.5, the sample of incorrectly predicted values and true labels of kannada MNIST dataset is represented.

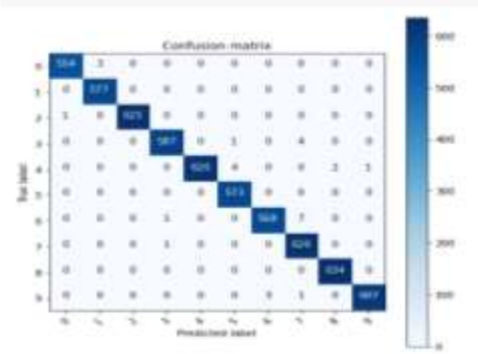

Fig 6.4: Confusion Matrix

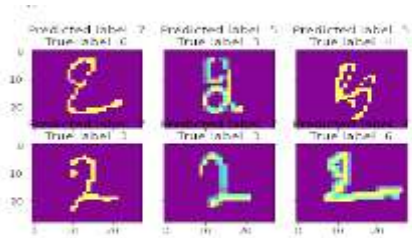

Fig 6.5: Predicted labels and true labels of MNIST dataset

The model is trained using the training data and check its performance on both the training and validation sets. The training average accuracy comes out to be 98.12% whereas the validation average accuracy is 98.95%. The Fig 6.6 represents the training and validation accuracy for 10 epochs.

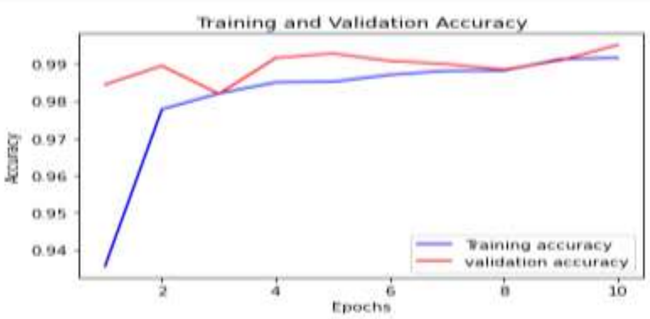

Fig 6.6: Training and Validation accuracy

The training loss indicates how well the model is fitting the training data, while the validation loss indicates how well the model fits new data. The Training and Validation loss graph is represented in Fig 6.7.

Fig 6.7: Training and Validation loss

IARJSET

ISSN (O) 2393-8021, ISSN (P) 2394-1588

International Advanced Research Journal in Science, Engineering and Technology

Vol. 8, Issue 11, November 2021

DOI: 10.17148/IARJSET.2021.81125

VII. CONCLUSION

In this work, The Kannada MNIST dataset has been used, Convolutional Neural Networks (CNN) & Recurrent Neural Networks (RNN) model are used for classifying the kannada handwritten digits has been proposed. The high presentation of image features are firstly extracts from the CNN, & then extracted features are used to recognize by the Gated Recurrent Unit. Initially, The CNNs confirms the greatest structures of the sample images that can be extracted using multi-convolutional layers & multi-pooling layers. Finally, Gated Recurrent Unit is a quick recognition classification method that establishes the consecutive association among features in hidden layers. The outcome of the result demonstrates that the proposed technique achieved 98.12% average training accuracy & 98.95% average validation accuracy & also achieved higher precision, recall & F1_score ratio.

For the future work, the plan is to adopt some other techniques by adding more LSTM network, implementing & use of CTC loss & using different activation function. It is expected to improve the performance of different architectures.

REFERENCES

[1] Baoguang Shi. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. CoRR. 2015.

[2] Youssouf C. Feature design and lexicon reduction for efficient off-line handwriting recognition. Ph.D. thesis. 2016.

[3] H. Sajedi, "Handwriting recognition of digits, signs, and numerical strings in persian," Computers Electrical Engineering, vol. 49, pp. 52– 65, 01 2016.

[4] M. H. Mirabdollah, M. A. Mohamed, and B. Mertsching, "Distributed averages of gradients (dag): A fast alternative for histogram of oriented gradients," Robot World Cup, pp. 97–108, 2016.

[5] S. Karthik and K. S. Murthy , "Deep belief network based approach to recognize handwritten kannada characters using distributed average of gradients," Cluster Computing, pp. 1–9, 2018.

[6] P. Bannigidad and C. Gudada, "Restoration of degraded Kannada handwritten paper inscriptions (Hastaprati) using image enhancement techniques," 2017 International Conference on Computer Communication and Informatics (ICCCI), Coimbatore, pp. 1-6, (2017)

[7] P. Yadav and M. Kumar, "Kannada character recognition in images using histogram of oriented gradients and machine learning," pp. 265– 277, 2018.

[8] Chherawala, Y., Roy, P.P., Cheriet, M.: Feature set evaluation for offline handwriting recognition systems: application to the recurrent neural network. IEEE Trans. Cybern. 46(12), 2825–2836 (2016)

[9] Jaderberg M. Reading text in the wild with convolutional neural networks. International Journal of Computer Vision. 2016.

[10] Roy, P., Ghosh, S., Pal, U.: A cnn based framework for unistroke numeral recognition in air-writing. In: 2018 16th International Conference on Frontiers in Handwriting Recognition (ICFHR). pp. 404{409. IEEE (2018).