Advances in Dynamical Systems and Applications.
ISSN 0973-5321, Volume 16, Number 2, (2021) pp. 1333-1347

https://www.ripublication.com/adsa.htm

Hand Written Character Identification of Kha Gunintham with Siamese CNN Network

B Soujanya[1]*, Ch Suresh[1] and T Sitamahalakshmi[1]

[1]Department of Computer Science and Engineering, Institute of Technology, GITAM (Deemedto be University), Visakhapatnam – 530045, India.

**Corresponding author.*

Abstract

Handwriting character recognition is an inspiring topic in the Telugu language due to the cursive nature of the handwriting and the similar shape of the characters. Lack of availability in the handwritten datasets related to Telugu characters has stretched the development of handwritten word recognizers and compared different methods in this field. It is challenging for modern deep neural networks that typically require hundreds or thousands of images per class. Learning prominent features of machine learning applications can be computationally expensive and proved that it would be difficult when a small amount of data is available. Recently, techniques such as one-shot learning, few-shot learning, and Siamese networks have been proposed to address this problem. The goal of this paper is to address the image similarity problem. We aim to get an image similarity function without using any manually designed features but instead directly learn this function from annotated pairs of raw image pairs. Inspiring by the advancement of deep learning, we decided to choose this technique in our experiments. Parameters are being updated across both the subnets, and then we find the similarity of the inputs. We achieved good results using a convolutional neural network that exceeded other deep learning models with comparable state-of-the-art performance on one- shot classification tasks.

INTRODUCTION

India is a multi-lingual multi-script nation that has more than 18 regional languages derived from 12 various scripts. Understanding the text written in handwritten documents enables access to the vast information in the scanned historical manuscripts. In recent years handwriting recognition has been one of the fascinating research domains in machine learning and image processing [1, 2].

Deep learning is an approach to machine learning that has experienced tremendous growth and success in recent years, mainly due to the availability of more powerful computers, larger datasets, and new techniques to train artificial neural networks.

Artificial Neural Network (ANN) is a paradigm for information processing inspired by the biological nervous system, composed of many highly interconnected neurons working to solve a specific problem. It is a deep learning algorithm similar to biological neural networks that accept type numeric and structured data input.

As the performance dropped in object detection and image classification problems, and with the accessibility of better CNN, the study of CNN in these areas has grown substantially [3-5], which explained various concepts and gave solutions in multiple ways. One of the influencing forms of ANN architecture is Convolution Neural Network (CNN).

Sunil Gundapu and Radhika Mamidi [6] implemented supervised learning to identify the domain of Telugu text data using the CNN-LSTM method and obtain 69% accuracy with test data. An accuracy of 96.5% has been achieved in identifying handwritten Telugu numerals using CNN – SVM by Durjoy Sen Maitra, Ujjwal Bhattacharya and Swapan K. Parui [7]. Still, the study is restricted to identify numerals only.

Hybrid CNN-RNN network [8] implemented by Minesh Mathew, Mohit Jain and C. V. Jawahar provided an accuracy of 57.2% only in recognizing Telugu scene images.

Konkimalla Chandra Prakash et al. [9] suggested the procedure which identifies the main character, vattu and gunintham separately using CNN with various architectures, providing a maximum accuracy of 98.6%.

A recent study by Tejasree Ganji et al. [10] using a pre-trained VGGNET model based on CNNarchitecture provided an accuracy of 92% in recognizing handwritten Telugu character datasetconsisting of 1600 characters.

The above studies are restricted to a few numerals or letters or implement restrictions in identifying handwritten Telugu characters. This shows a need to develop a model that can recognize handwritten Telugu characters and numerals with higher accuracy.

Similarity learning is an area of supervised machine learning. The goal is to learn a similarity function that measures how similar or related two objects are and returns a similarity value. A higher similarity score is returned when the objects are identical, and a lower similarity score is returned when the objects are different. All the natural language problems can be solved by similarity learning. For finding the similarity, we use the Siamese neural network.

Siamese Neural Network is often widely used for author verification [11, 12], forgery offline signature verification problems [13], OCR [14] with better accuracy. Until to date, only one work has been done on recognition of Telugu handwritten characters using Siamese network with two CNNs by D T Mane et al., [15]

SIAMESE CNN Network:

Neural networks are good at every task in the recent deep learning epoch, but these neural networks to perform well need more data.

In the recent Deep learning epoch, Neural networks are almost good at every task, but neural networks depend on more data to perform well. But for some applications like face recognition and signature verification, it does not always rely on getting more data. A new-fangled type of neural network architecture called Siamese Networks is used to solve this kind of task.

The Siamese network uses very few numbers of images to get improved predictions. The popularity of Siamese networks in recent years is due to the ability to learn from very little data. We will sightsee what this article is and how to develop a system to identify the characters in agunintham with Pytorch using Siamese Networks.

The images belonging to the same class or different class will be given input to this network. Output 1 indicates that the two images are of the same class and 0 represents dissimilar classes.[17, 18]

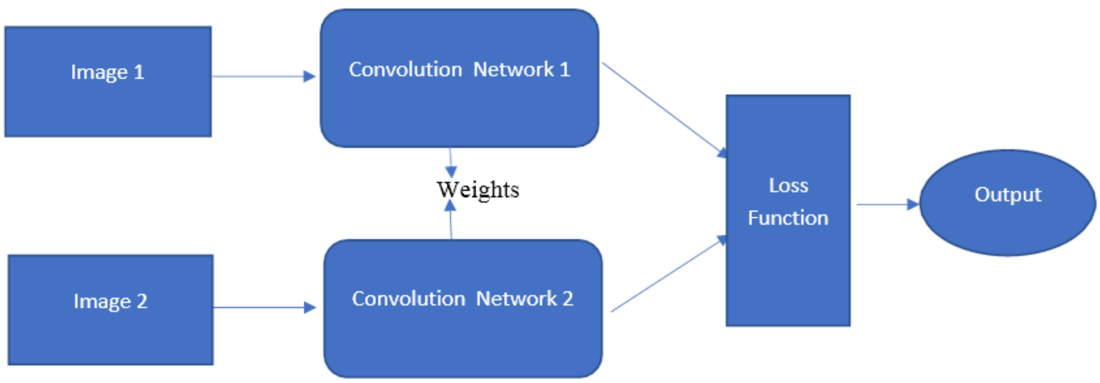


Figure 1: Structure of Siamese neural network

There are two identical sub neural networks with the same weights.

- From the pair of images, each image is fed to one of these networks.
- Contrastive loss function is used to optimize the networks.

Contrastive Loss function:

The main intention of Siamese architecture is not to categorize input images but to differentiate between them. This architecture is well suited to use a contrastive function. Instinctively, the contrastive function is used to evaluate how well the Siamese CNN distinguishes image pairs.

The representation of contrastive loss function is:

$$(1-Y)\frac{1}{2}(D_W)^2 + (Y)\frac{1}{2}\{max(0, m - D_W)\}^2 \quad \text{.......................................(1)}$$

The *Dw* is the Euclidian distance between sister Siamese networks between the outputs Mathematical form of Euclidean distance is :

$$\sqrt{\left\{G_w(X_1) - G_w(X_2)\right\}^2} \quad \text{.......................................(2)}$$

The input data pairs are X1 and X2, Whereas *Gw* is the output of one of the sister networks.

Data Preparation

1. Load train data images from the root folder which has all images of **Ka guninthalu**
2. Define the categories all words in Ka gunintham.
3. Assign a respective category to each image based on the image.
4. Resize and reshape all the images with fixed LENGTH * BREATH
5. Convert all the images to 3-dimensional Numpy array and store it for future to inputto train model.
6. Similar steps should be followed for test images and save it for the future to test thesimilarity between images from model

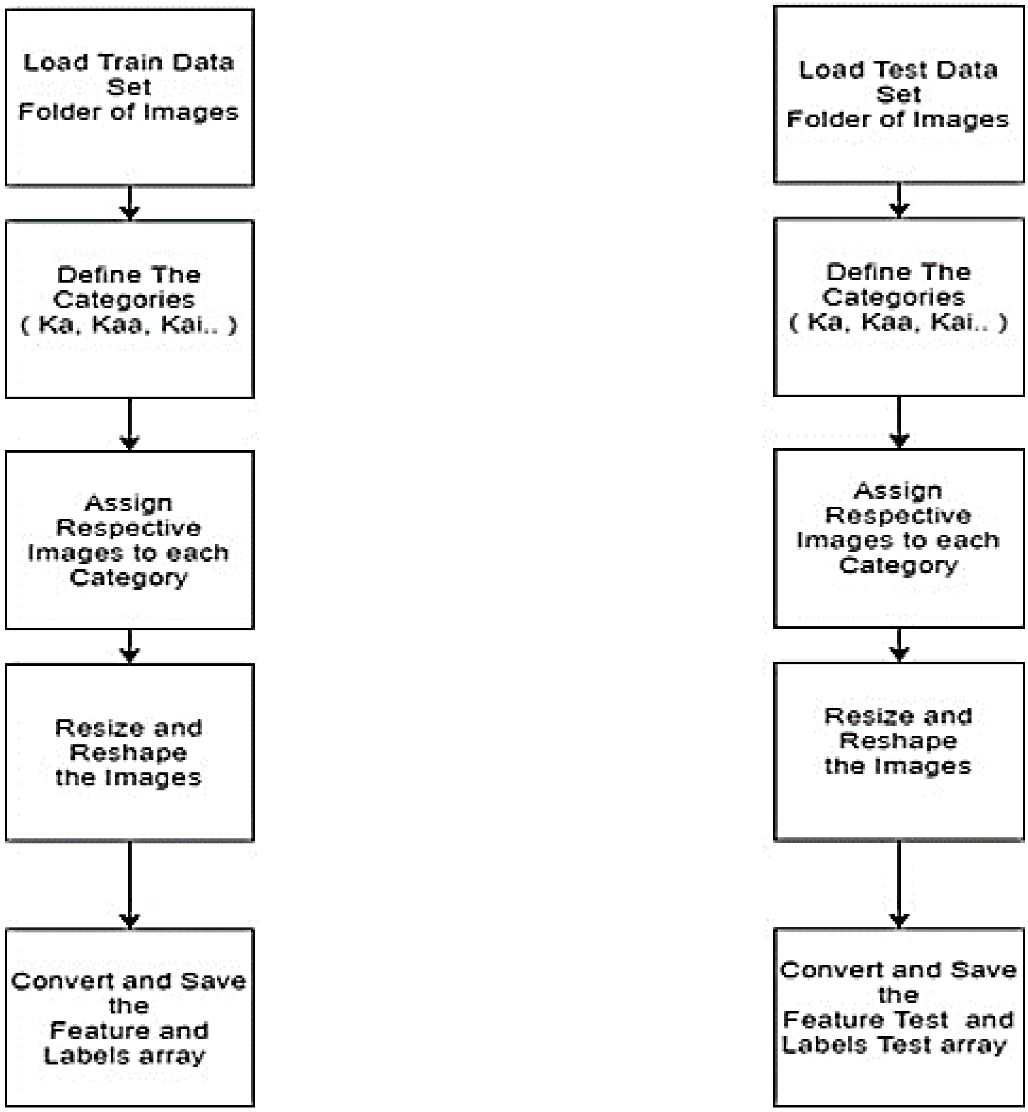


Figure 2: Features and Labels for the data set

Train Model

1. Load the train and test data which we save in the previous step
2. Make positive and negative pairs from trainX and trainY as [**pairTrain, labelTrain**]
3. Make positive and negative pairs from testX and testY as [**pairTest, labelTest**]
   a. **Preparation of positive and negative pairs**
      i. To indicate a positive or negative pair by initializing two empty liststo hold the (image, image) pairs and labels.
      ii. Lit of indexes built for each class provides the indexes for all the examples

with a label after calculating the total number of classes present in the dataset.

iii. loop over all images

(1) randomly pick an image that belongs to the same class, grab the current image and label belonging to the current iteration

(2) update the images and labels after preparing a positive pair.

(3) randomly pick an image corresponding to a label not equal to the current label and grab the indices for each of the class labels not identical to the current label

(4) update our lists subsequently preparing a negative pair of images

iv. to return a 2-tuple of our image pairs and labels

4. Building Siamese network

5. Initialize the input Image Size of two images ImgA, IMDb

a. stipulate the inputs for the feature extractor network.

b. describe the first set of CONV => RELU => POOL => DROPOUT layers and second set of CONV => RELU => POOL => DROPOUT layers.

c. prepare the final outputs

d. build the model

e. return the model to the calling function

6. extract features of ImgA and ImgB and assign to and featsA and featsB

7. finally, construct the siamese network

a. find the euclidian distance of two images

b. prepare output features as dense 16 (size of gunithalu)

c. Construct Model (image, image, outputs)

8. compiling model with loss as sparse categorical cross-entropy and optimizer as adam and metrics as accuracy.

9. Train the model with training data as Train features and Validation data as Test features

10. Save the Model for future testing.

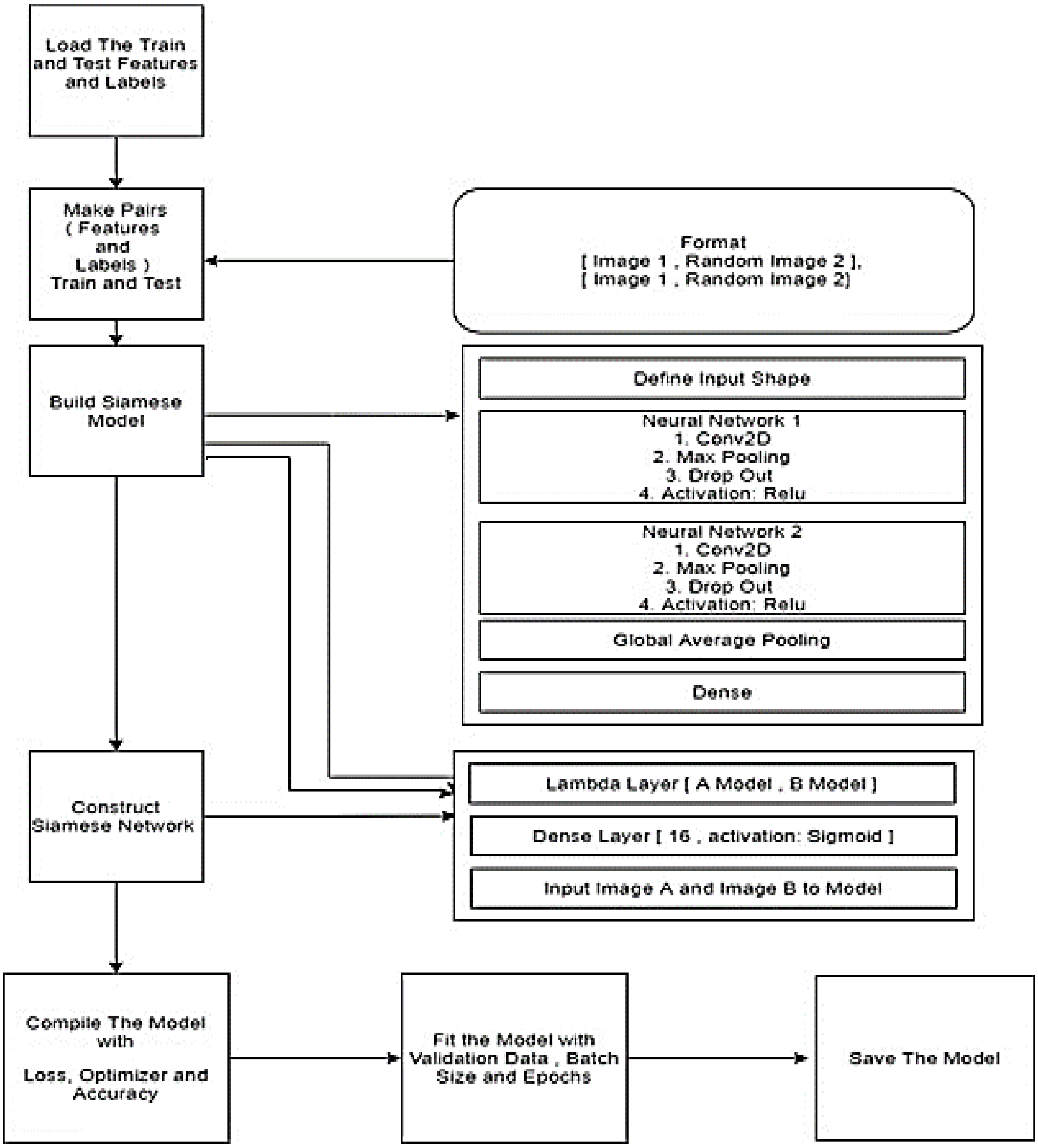


Figure 3: Flow of Siamese Network to train the data.

Test Model

1. Load the train model for testing
2. Iteratively compare each image with the rest by sending two images as parameters to predict the model and then calculating the percentage by multiplying to 100.
3. Test same for all images and store the scores

4. Finally save the data into CSV file

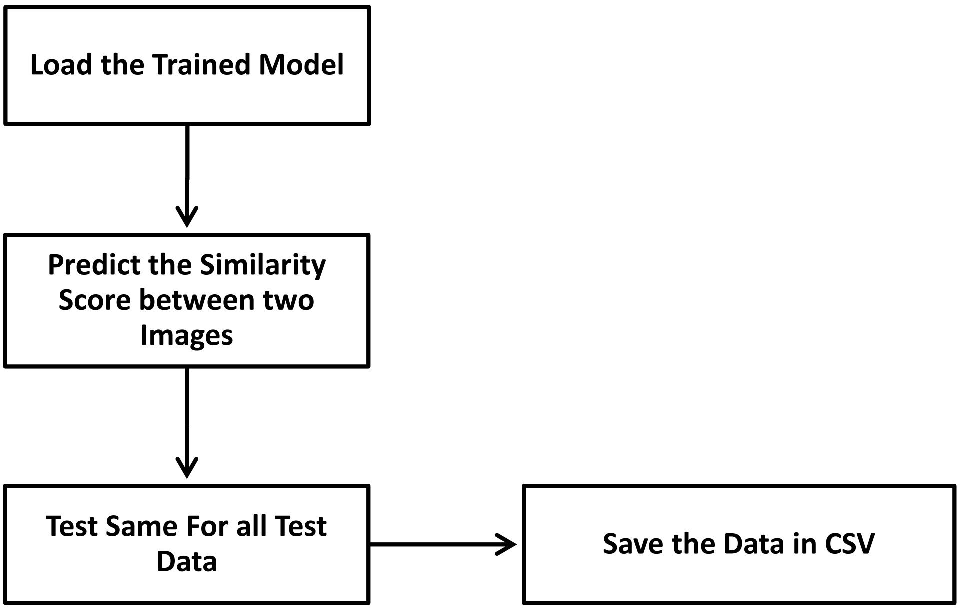


Figure 4: Test Siamese Network

Siamese neural network, also termed a twin neural network, consists of two or more identical subnetworks with the same configuration and co figuration for other sub-networks. We feed a pair of inputs to these networks where each network computes the features of one input, and then the similarity is calculated using their difference or the dot product. The last layers of the two networks are then nourished to a contrastive loss function, which calculates the similarity between the two images.

RESULTS & DISCUSSIONS

The results discussed in this section shows how the Siamese CNN network estimates the similarity between the images. Here we trained the model with nearly 32 differently written sets of Kha gunintham. We tested with nearly 16 differently written sets and the results are presented from tables 2 to 4. As there is no pre-defined data set is available, we created our own dataset with nearly 64 differently written kha gunintham characters. The diagonal in the matrices represent the similarity index, and in almost all the tested models, the diagonal valueis 62.4, it is above 50 means the model finds the similarity between the tested and trained kha gunintham characters exactly. So, the accuracy obtained with the proposed Siamese CNN based character identification for all the characters of KHA gunintham is 100 percent.

Table 1: The notations used for the proposed model are

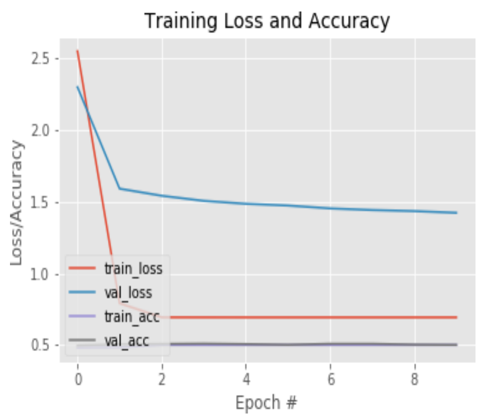
Notationused	Indexvalue	Originalnotation
Ka	0	
Kaa	1	
Kaha	2	
Kai	3	
Kam	4	
ke	5	
kee	6	
ki	7	
kii	8	
ko	9	
koo	10	
kou	11	
kru	12	
kruu	13	
Ku	14	
kuu	15	

The Siamese model predicted the following results with different test data images concerning differently written characters of kha gunintham.

Table 2: Sample outputs and plots for few Siamese Similarities for index values 1,2,3,4,5 respectively.

	ka	kaa	kaha	kai	kam	ke	kee	ki	kii	ko	koo	kou	kru	kruu	ku	kuu
0	62.4	0	0	0	0	1.7	0	0	0	0	0	0	0	0	0	0
1	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0.1
3	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0
5	1.7	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	62.4	0	0	0	0.3	0	0	0	0	0.1
7	0	0	0	0	0	0	0	62.4	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	62.4	0	7.3	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	3.6	0
10	0	0	0	0	0	0	0.3	0	7.3	0	62.4	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0
14	0	0	0	0	0	0	0	0	0	3.6	0	0	0	0	62.4	0
15	0	0	0.1	0	0	0	0.1	0	0	0	0	0	0	0	0	62.4

	ka	kaa	kaha	kai	kam	ke	kee	ki	kii	ko	koo	kou	kru	kruu	ku	kuu
0	62.4	0	0	0	0	0	0	0	0	0	0	15	0	0	0	0
1	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	62.4	0	0	0	0	0.2	0	0	0	0	0	0	0	0
3	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	62.4	0	0	0	0.1	0	0	0	0	0	0	0
5	0	0	0	0	0	62.4	7.4	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	7.4	62.4	0	0	0	0	0	0	0	0	0
7	0	0	0.2	0	0	0	0	62.4	0	0	0	0	0	0	0	0
8	0	0	0	0	0.1	0	0	0	62.4	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	0
11	15	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4



	ka	kaa	kaha	kai	kam	ke	kee	ki	kii	ko	koo	kou	kru	kruu	ku	kuu
0	62.4	0	0	0	0	23.1	0	0	0	0	0	0	0	0	0	0
1	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0
5	23.1	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	62.4	0	0	0	0	0	0	0	3.2	0
7	0	0	0	0	0	0	0	62.4	0	7.4	0	0	0	0	0	0.5
8	0	0	0	0	0	0	0	0	62.4	0.1	0	0	0	0	0	0
9	0	0	0	0	0	0	0	7.4	0.1	62.4	0	0	0	0	0	0.1
10	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0
14	0	0	0	0	0	0	3.2	0	0	0	0	0	0	0	62.4	0
15	0	0	0	0	0	0	0	0.5	0	0.1	0	0	0	0	0	62.4

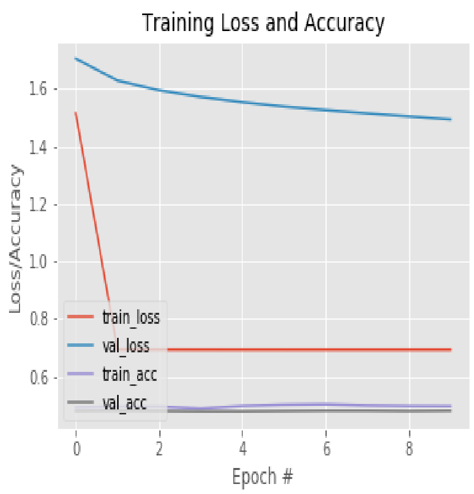
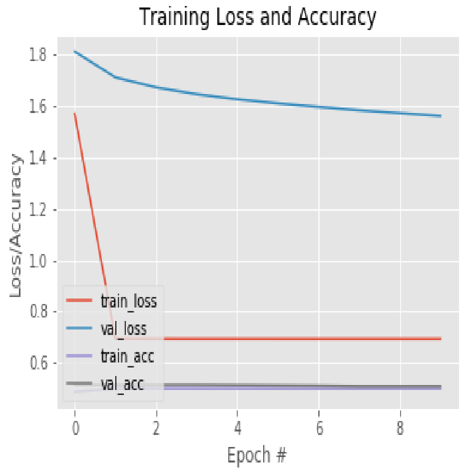
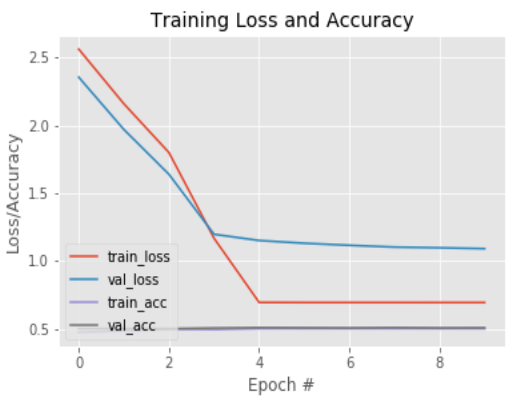


Table 3: Sample outputs and plots for few Siamese Similarities for index values 1 ,2,3,4,5respectively.

	ka	kaa	kaha	kai	kam	ke	kee	ki	kii	ko	koo	kou	kru	kruu	ku	kuu
0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	62.4	0	0	0	0.1	0	0	0	0	0	1	0	0	0	
2	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	
3	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	
5	0	0.1	0	0	0	62.4	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	62.4	0	0	0	0	0	0	0	0	
7	0	0	0	0	0	0	0	62.4	0.5	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	0.5	62.4	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	
11	0	1	0	0	0	0	0	0	0	0	0	62.4	0	0	0	
12	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	
13	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	
15	0	0	0	0	0	0	0	0	0	0	3.3	0	0	0	0	6



	ka	kaa	kaha	kai	kam	ke	kee	ki	kii	ko	koo	kou	kru	kruu	ku	kuu
0	62.4	0	0	0	0	5.9	0	0	0	0	0	0	0	0	0	
1	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0	
2	0	0	62.4	0	0	0	0	0	0	0	0	6.6	0	0	0	
3	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	62.4	0	0	0	0	0	3.8	0	0	0.6	0	
5	5.9	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	62.4	0	0	0	0	0	0	0	0	
7	0	0	0	0	0	0	0	62.4	0	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	0	62.4	0	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	0	
10	0	0	0	0	3.8	0	0	0	0	0	62.4	0	0	0	0	
11	0	0	6.6	0	0	0	0	0	0	0	0	62.4	0	0	0	
12	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	
13	0	0	0	0	0.6	0	0	0	0	0	0	0	0	62.4	0	
14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	



	ka	kaa	kaha	kai	kam	ke	kee	ki	kii	ko	koo	kou	kru	kruu	ku	kuu
0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
1	0	62.4	0	0	0	0	0	0	0	0	0	0	0	0	0.1	
2	0	0	62.4	0	0	0	0	0	0	0	0	0	0.7	0	0	
3	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	0	
4	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	0	
5	0	0	0	0	0	62.4	0	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	62.4	1.6	0	0	0	0	0	0	0	
7	0	0	0	0	0	0	1.6	62.4	0	0	0	0	0	0	0	
8	0	0	0	0	0	0	0	0	62.4	27.4	0	0	0	0	0	
9	0	0	0	0	0	0	0	0	27.4	62.4	0	0	0	0	0	
10	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	0	
11	0	0	0	0	0	0	0	0	0	0	0	62.4	0	0	0	
12	0	0	0.7	0	0	0	0	0	0	0	0	0	62.4	0	0	
13	0	0	0	0	0	0	0	0	0	0	0	0	0	62.4	0	
14	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	62.4	
15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

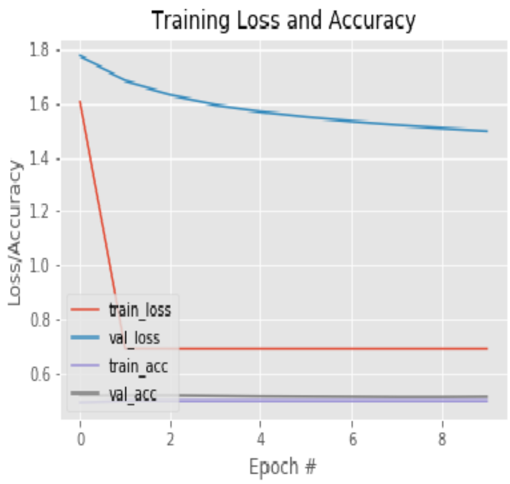


Table 4 shows the values of Accuracy obtained with 16 different variations test data of kha gunintham characters. The similarity value was measured with 32 different variations training data of kha gunintham characters. The similarity value obtained will results the exact identification of the characters of kha gunintham. As the test image is exactly identified with training image so the accuracy for all the test samples is 100.

Table 4: The accuracy measured with Siamese Similarities for index values 1,2,3,4,5 respectively.

Training Set	Accuracy obtained with Siamese Network in %															
	Ka	Kaa	Kaha	Kai	Kam	Ke	Kee	Ki	Kii	Ko	Koo	Kou	Kru	Kruu	Ku	Kuu
1	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
2	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
3	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
7	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
8	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
9	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
10	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
11	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
12	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
13	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
14	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
15	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
16	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100

Table 5 depicts the value of the Contrastive Loss function of test character of kha gunintham with the different training characters of kha gunintham. The loss will be ‘zero’ for positive pairs,which means the representation of both the characters with no distance between them and lossincreases with an increase in the distance.

Table 5: The Contrastive Loss function measured with Siamese Similarities for index values1,2,3,4,5 respectively.

Training Set	Loss Function with Siamese Network															
	Ka	Kaa	Kaha	Kai	Kam	Ke	Kee	Ki	Kii	Ko	Koo	Kou	Kru	Kruu	Ku	Kuu
1	0	0	0	0	0.66	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0.19	0	0	0	0	0	0.23	0	0	0	0	0	0
4	0	0	0	0	0.22	0	0	0	0	0	0	0	0.21	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.32
6	0	0	0	0	0	0	0	0	0	0	0	0.18	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	0.36	0	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0.45	0	0	0	0	0	0	0	0.71	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0	0.62	0	0
14	0	0	0	0	0	0.22	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0.33	0	0	0.19	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

CONCLUSION

Instead of using huge data for each class, the similarity between two classes can be calculated.The output will always be a floating-point number extending between 0 and 1. The two images belong to the same class indicated by 1, whereas 0 represents the images are from different classes. The Siamese Neural Network used to identify the similarity between the characters of kha gunintham proposed in this work produces the similarity index for all the tested images withtraining examples with a value of 62.4 for the matched images and a value 0 or less than 50 forthe images which are not matched. From the results, the Siamese CNN network is most suitablefor character identification. From the above results, we can conclude that the proposed method is suitable not for kha gunintham and all the other gunintham in Telugu literature.

REFERENCES

[1] S. Mori, C.Y. Suen and K. Kamamoto, 1992 “Historical review of OCR research and development,” Proc. of IEEE, vol. 80, 1029-1058,

[2] S. Impedovo, L. Ottaviano and S. Occhinegro, 1991 “Optical character recognition”, International Journal Pattern Recognition and Artificial Intelligence, Vol. 5(1-2), 1-24

[3] Srinivas, S., Sarvadevabhatla, R. K., Mopuri, K. R.,Prabhu, N., Kruthiventi, S. S., &Babu, R. V. (2016) “A taxonomy of deep convolutional neural nets for computer vision.” https://doi.org/10.3389/frobt.2015.00036

[4] Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., &Torralba,A. (2014) “Object detectorsemerge in deep scene cnns.” https://arxiv.org/abs/1412.6856

[5] Wang, Y., & Wu, Y. “Scene Classification with DeepConvolutional Neural Networks.”(2014) https://yzhwang.github.io/ecs289h-vision.pdf

[6] Sunil Gundapu, Radhika Mamidi "Multichannel LSTM-CNN for Telugu Technical Domain Identification" (2021) arXiv:2102.12179 https://arxiv.org/pdf/2102.12179.pdf

[7] Maitra, Durjoy Sen, Ujjwal Bhattacharya, and Swapan K. Parui. "CNN based common approach to handwritten character recognition of multiple scripts." *2015 13th International Conference on Document Analysis and Recognition (ICDAR)*. IEEE, 2015.

[8] Mathew, Minesh, Mohit Jain, and C. V. Jawahar. "Benchmarking scene text recognition in Devanagari, Telugu and Malayalam." *2017 14th IAPR International Conference onDocument Analysis and Recognition (ICDAR)*. Vol. 7. IEEE, 2017.

[9] Prakash, Konkimalla Chandra, et al. "Optical character recognition (OCR) for Telugu: Database, algorithm and application." *2018 25th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2018.

[10] Ganji T, Velpuru MS, Dugyala R. Multi Variant, 2021 “Handwritten Telugu Character Recognition Using Transfer Learning” IOP Conference Series: Materials Science and Engineering Vol. 1042, 012026

[11] Adak, Chandranath, et al. "Offline Bengali writer verification by PDF-CNN and Siamese net." *2018 13th IAPR International Workshop on Document Analysis Systems(DAS)*. IEEE, 2018

[12] Dlamini, Nkosikhona, and Terence L. van Zyl. "Author Identification from Handwritten Characters using Siamese CNN." *2019 International Multidisciplinary Information Technology and Engineering Conference (LIMITED)*. IEEE, 2019

[13] Victoria Ruiz, Ismael Linares, Angel Sanchez, Jose F. Velez "Offline handwritten signature verification using compositional synthetic generation of signatures and Siamese Neural Networks" Neurocomputing, 374 (2020) Pages 30-41

[14] Sokar, Ghada, Elsayed E. Hemayed, and Mohamed Rehan. "A generic OCR using deep siamese convolution neural networks." *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (ICON)*. IEEE, 2018.

[15] Mane, D. T., and Uday V. Kulkarni. "A survey on supervised convolutional neural network and its major applications." *Deep Learning and Neural Networks: Concepts, Methodologies, Tools, and Applications*. IGI Global, 2020. 1058-1071.

[16] https://towardsdatascience.com/a-friendly-introduction-to-siamese-networks-85ab17522942

[17] https://towardsdatascience.com/siamese-networks-line-by-line-explanation-for-beginners-55b8be1d2fc6