Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

Holistic Approach Employing Different Optimizers for the Recognition of District Names Using CNN Model

Sandhya Sharma[1], Sheifali Gupta[2], Neeraj Kumar[3]

[1,3]Chitkara University Institute of Engineering and Technology,
Chitkara University, Himachal Pradesh, India

[2]Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India

[1)]sandhya.sharma@chitkarauniversity.edu.in

Abstract: A Holistic approach is proposed for the recognition of Handwritten district names of Punjab state which are written in Gurmukhi Script. For the purpose of recognition, a Convolutional Neural Network(CNN) using deep learning is employed. Initially, the dataset of 22000 of images is prepared for all the 22 district names of Punjab state and later a CNNis employed. The proposed CNN architecture having 12 layers is developed and employed using three optimizers: Adam, SGD and RMSprop for the recognition task. Best Average Validation Accuracy achieved for the proposed CNN architecture is 95% and maximum achieved validation accuracy is 99% achieved by employing Adam Optimizer.

Keywords: Word Recognition, Convolutional Neural Network, Gurmukhi Words, Holistic approach, Postal Automation

1. INTRODUCTION

With the advancement of technology, Artificial Intelligence (A.I): Machine learning and Robotics, this is the time for robots to perform all the operations similar to human intelligence. Machines use their own senses to perform various kinds of task like image recognition, pattern recognition, interpreting the natural language and learning and solving the various problems etc. Handwriting identification plays a significant role in the era of digitization. The main purpose of handwriting recognition is to convert the handwritten text into machine readable format so that it can be further processed and used in various applications like automatic reading of car license plates, preservation of handwritten historical documents, automation of old official documents and most importantly for the automation of postal system. All these areas deal with huge amount of data so they need a system which can provide good recognition accuracy with less complexity and persistent performance for the purpose of recognition. One such system is possible which is Convolutional Neural Network(CNN) and is based on Deep learning. Deep learning models generates good recognition results when dealing with huge amount of data by extracting high level of abstract features. Recognition of handwritten text is a difficult task in computer vision and machine learning. Uncertainty of input data makes it difficult to recognize handwritten text because handwriting is different for every person, some characters may be distorted, some may have similar shape and also different scanners used for the scanning of text may also generate uncertainty in the data.

Postal Automation is one of the potential application areas in the field of text recognition. Postal automation is helpful in the automatic handling or sorting of mails. As Mail-handling

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

is a very labor intensive process and labor costs have been increasing during the last three decades. Automatic reading is necessary for all the address fields necessary for the carrier to reach the final destination. Several postal automation systems are available for the nations like Australia, USA, Canada, France and UK. . There is currently no postal automation system exist for Gurmukhi script while it is available in various other scripts like Bangla, English and Tamil etc. This proposed work will be helpful in Postal automation in Gurumukhi script.Gurumukhi being the official language of the Punjab(state) in North India and has been accorded the status of official language by the Punjab government for the administrative task. Because of this, the address field of all the postal documents is usually written in Gurmukhi script. There is currently no postal automation system exist for Gurmukhi script. So, for the recognition of handwritten district names of Punjab, a deep learning model is proposed to recognize all the 22 district names of Punjab(state) which are written in Gurmukhi script.

2. RELATED WORK

In the literature, two approaches are used for the recognition of text: Analytical approach [1,2] and Holistic[3]. In Analytical approach, complete word is divided into individual characters and then recognition task is carried out while in Holistic approach, recognition is carried out on complete word without segmenting it into individual characters. It is very difficult to find segmenting points in a script like Gurmukhi which is cursive in nature. So, In the proposed work, Holistic approach is implemented. Bhowmik et al[4] has employed Holistic approach for the characters recognition of Bangla handwritten city names. Various features like tetragonal, elliptical and vertical pixel density histogram are extracted. For the purpose of recognition Support vector machine(SVM and) Multi-layer perceptron(MLP) are employed. Daniyar et al[5] has carried a recognition of city and country names written in Russian and Kazakh languages using Deep learning models and achieved an accuracy of 55.3% using CNN. As India follows the multilingual behavior so Pal et al[6] has proposed a work on the recognition of Multi-lingual(Hindi, English and Bangla) script based city names for Indian postal Automation. Firstly the slant correction technique is employed for taking care of the slanted handwriting and then water reserviortechnique is employed for their segmentation into individual characters. Once the characters are obtained, features are extracted and dynamic programming is employed for their recognition. Overall obtained accuracy is 92.5%. Similarly Thadchanamoorthy et al[7] has also performed the recognition of Tamil city names in which city names are divided into individual characters and later merged into possible characters using dynamic programming based on Modified quadratic discriminant function(MQDF) and then features are extracted and recognized. Accuracy obtained is 96.89%. It can be observed [6],[7] that manual feature extraction is carried out before recognizing the city names which is a cumbersome process. So, in the proposed work of district names recognition automatic feature extraction using CNN is carried out on the complete word without segmenting it into individual characters. Vajda et al[8] has proposed a postal automation system for the city names and pin codes written in English and Bangla and achieved an accuracy of 86.44% for city names written in Bangla while 93% for English numerals and 94.13% for Bangla numerals with the help of Non-symmetric half plane hidden markov model. Features are manually extracted but recognition is carried out using

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

Holistic approach. Pal et al[9] has proposed a similar approach[8] for the recognition of Bangla script written street names and obtained an accuracy of 99.03%. Techniques for the recognition of Pincode numerals which are written in Bangla are discussed in paper [10][11]. For the purpose of sorting the postal documents which are written in Bangla and Arabic, Roy et al[9] has proposed a two-stage Multilayer perceptron classifier and achieved an accuracy of 92.10%.Full pin code recognition for 6 digit string is carried out[11], pre-segmentation, feature extraction and recognition is done using dynamic programming and accuracy obtained is 99.01%.

3. METHODOLOGY

In the proposed methodology, the first step is to prepare the dataset for all the district names of Punjab state followed by digitization and preprocessing and later a CNN model is designed for the purpose of recognition.

3.1 Dataset

For the creationof Gurmukhi handwritten words of 22 district names, 500 different writers wrote each word 2 times, resulting in total 22000 wordswheras, the writers were considered from different age groups, different educational and professionalqualifications.

3.2 Digitization and Pre-processing

Once the dataset is collected, it is converted into digital form using a scanner and some pre-processing is employed like binarization and normalization. The binary form of the image is obtained in binarization, comprising blackand white pixels, and each word is sliced one by one from the binarized image in Normalization. Few images of the prepared dataset is shown in Figure 1.

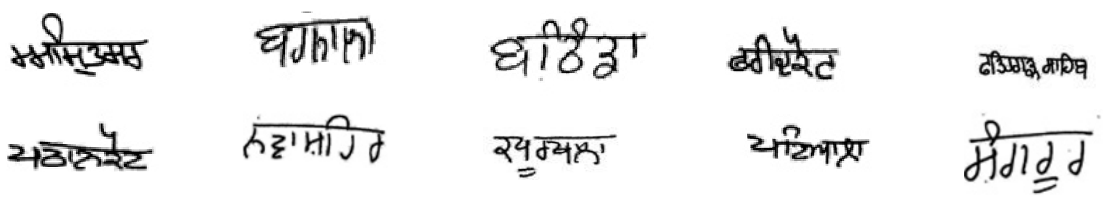


Figure 1. Few Images of prepared dataset

3.3 Processing

In this step ConvNet is designed having 12 layers. CNN model has three main layers[12]: convolution, pooling and fully connected layers as per the diagram shown in Figure 2. The main purpose of convolution and pooling layer is to extract the abstract features while the output layer which is the fully connected layer, helps in delivering the output in the form of output classes.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

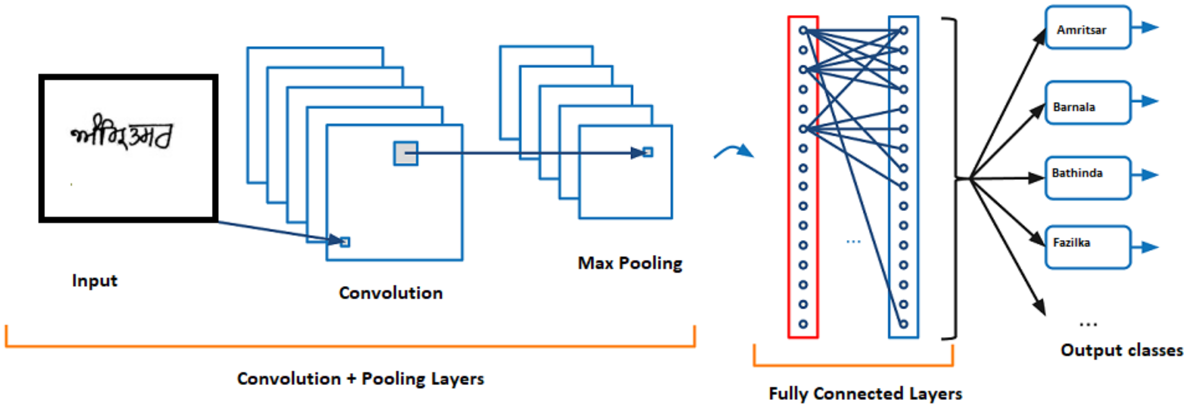


Figure 2 . Overview of CNN model

The proposed model with 12 layers is shown in Figure 3. Firstly, the input image of size 64x32x3 is given to the first convolution layer which has 32 filters of size 3x3 and a stride of 1x1, from which 32 feature maps are obtained which is followed by ReLU activation function. Next MaxPooling layer is introduced having pool size of 2*2 and stride of 2x2, it means that size of feature map is reduced by a factor of 2 using Maxpooling layer. Second convolution layer is introduced having 64 filters of size 3x3 and a stride of 1x1 which is followed by max-pooling layer of size 2x2 and a stride of 2x2 which is again followed by another convolution layer having 128 filters of size 3x3 and again followed by maxpooling layer. The last convolution layer is added having 256 filters of size 3x3 which is further followed by maxpooling layer. Lastly fully connected layer is introduced which has 2048 neurons in the input layer, 120 neurons in the middle layer and 22 neurons in the output layer.

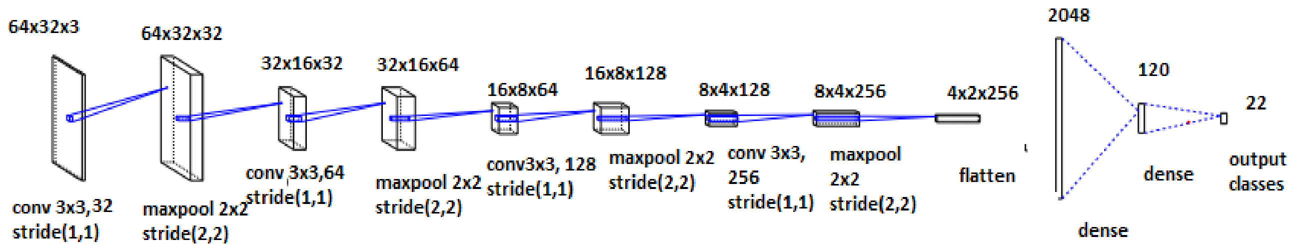


Figure 3 . Proposed model with layers and their filter sizes

4. EXPERIMENTS AND RESULTS

4.1 Model Training

The proposed model is evaluated on the newly created dataset which contains total of 22000 handwritten word images for 22 different classes, where each class has 1000 samples. Data is randomly split into training and validation dataset where 80% of the data is kept for training of the model and 20% for validating the model[13] out of 1000 collected samples of each class. The statistics of dataset used for performing the experiment is given in Table1.Size of the input image for the purpose of training and validation used is 64x48x3

Gurmukhi handwritten city name statistics used

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

Table 1. Statistics used for the experimentation

Dataset	Training dataset	Validation dataset	Total dataset samples	Number of classes
Gurmukhi handwritten district names of Punjab for all the 22 districts	17,600	4400	22000	22

The model is trained using two different optimizers which are named as Adam, SGD and RMSprop. The training parameters for the proposed work are given below.

- Optimizer: Adam, SGD and RMSprop
- Learning rate: 0.001
- Number of epochs: 45
- Dropout: 0.5
- Number of layers: 12

Parameters specifications of the proposed model is shown in Table 2. It describes the size of input image, size of filter used, number of filters, Activation function, size of output image with Number of parameters for each layer of the proposed model. The Number of parameters are obtained by using the relation:No. of parameters = Number of filters in the current layer * (number of filters in the previous layer (width of filter *height of filter) * +1) while number of parameters for dense layer are obtained by using the relation: Number of parameters = (neurons in the current layer * neurons in the previous layer)+ 1* neurons in the current layer

Table 2. Parameters specifications of the proposed model

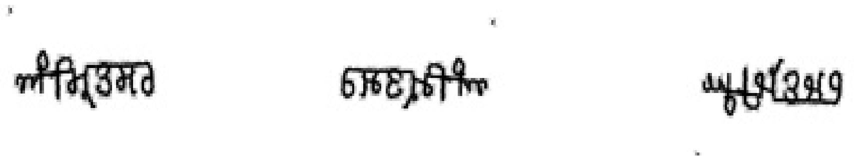
Layers	Input Image size	Filter size	No of filters	Activation function	Output without padding	Parameters
Input image	64*32*3					
Convolution	64*32*3	3*3	32	ReLU	64*32*32	896
MaxPooling	64*32*32	Pool size(2*2)	----------	----------	32*16*32	0
Convolution	32*16*32	3*3	64	ReLU	32*16*64	18496
MaxPooling	32*16*64	Pool size(2*2)	----------	----------	16*8*64	0
Convolution	16*8*64	3*3	128	ReLU	16*8*128	73856
MaxPooling	16*8*128	Pool size(2*2)	----------	----------	8*4*128	0
Convolution	8*4*128	3*3	256	ReLU	8*4*256	295168
MaxPooling	8*4*256	Pool size(2*2)	----------	----------	4*2*256	0
Flatten	4*2*256	----------	----------	----------	2048	-----------

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

Dense	2048	120	----------	ReLU	120	245880
Dense	120	-----------	----------	SoftMax	22	2662

4.2 Execution

The proposed model is executed using Tensorflow and keras which are open source machine learning libraries[14]. Tensorflow is mainly deep learning library while Keras is Neural Network library. For the purpose of training and validating the model, Google Collaboratory is used which is free cloud service used for developing deep learning applications. For obtain the more transformed images of the original data, data augmentation is used using ImageDataGenerator which is a class of Keras library. Data augmented images are shown in Figure 4 which are horizontally and vertically flipped.



(a) (b) (c)

Figure 4: Data Augmentation: (a) Original image (b) Horizontal flipped image (c) Vertical flipped image

4.3 Results and Discussions

In this section, the proposed model is analyzed using three Optimizers namely Adam[15], SGD[16] and RMSprop[17]. The model is evaluated on 45 epochs with a batch size of 4 and learning rate of 0.001. Number of epochs plays an important in the training of the model which helps in increased accuracy. It can also be concludedthat small batch sizes provides better training of the model and hence good recognition accuracy[18].

4.3.1 Recognition results achieved by using Adam Optimizer

Parameters convergence plot obtained using Adam optimizer is shown in Figure5. It can be observed from Figure 5(a) that validation loss is highest on 3rd epoch but it started decreasing after that for each epoch while the plot for training loss is linearly decreasing. In Figure 5(b), the validation accuracy is least on 3rd epoch but after that it is almost increasing. Similarly in Figure 5(c) and 5(d), the validation recall and precision is least on 3rd epoch, but after that the values are approaching to 1.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

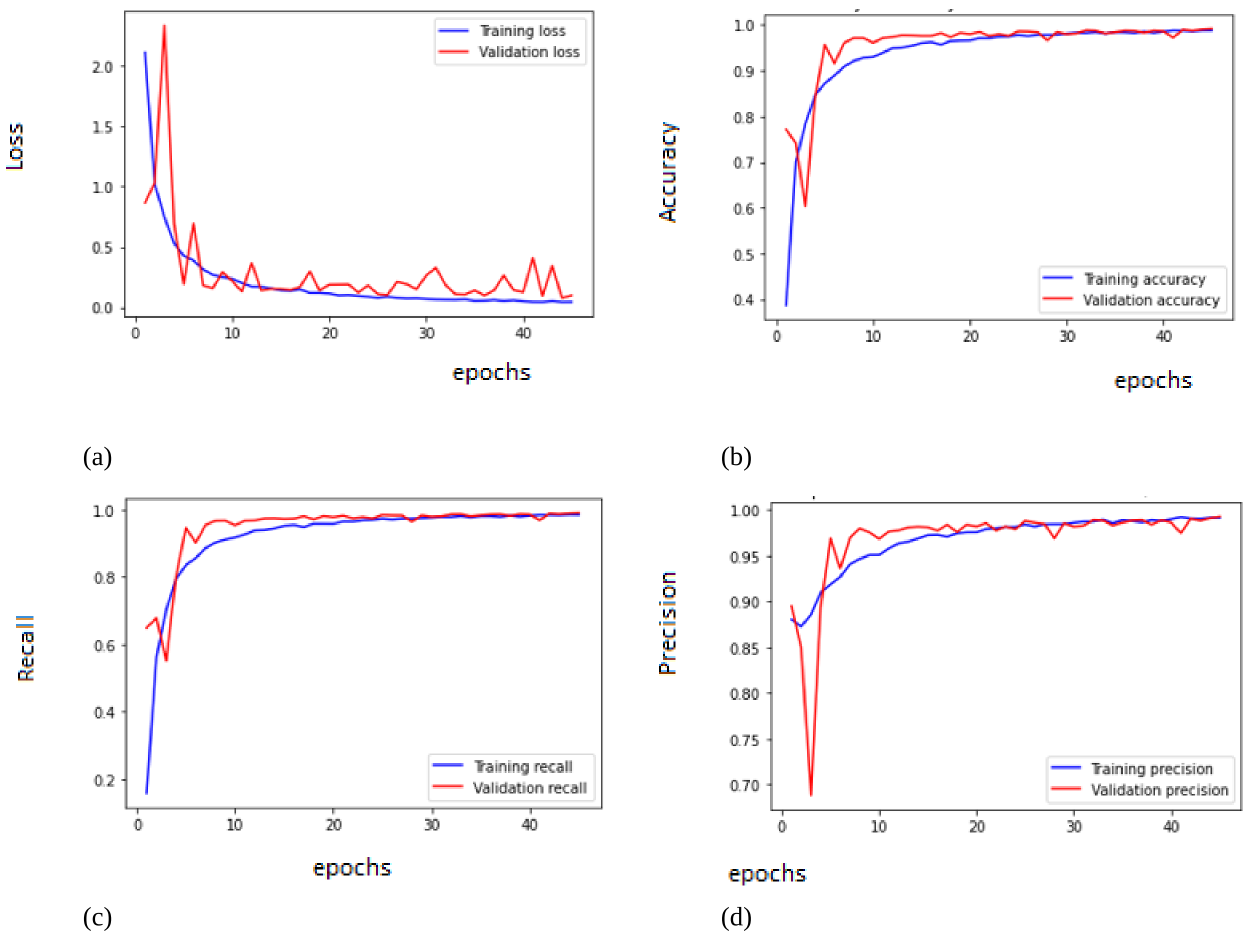


Figure5. Parameters convergence plot obtained using Adam optimizer with a learning rate of 0.001 (a) Training and validation loss, (b) Training and validation accuracy, (c) Training and validation recall, (d) Training and validation precision

4.3.2 Recognition results achieved by using SGD Optimizer

Now, the optimization is done using SGD optimizer. Parameters convergence plot for the same is shown in Figure 6.In Figure 6(a), maximum training loss and validation loss on the 1^{st} epochis around 2.5 and 1.5 and later started decreasing for each epoch. From the Figure 6(b), 6(c) and 6(d), it can be observed that the validation accuracy, recall and precision are almost linearly varying after 25^{th} epoch.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

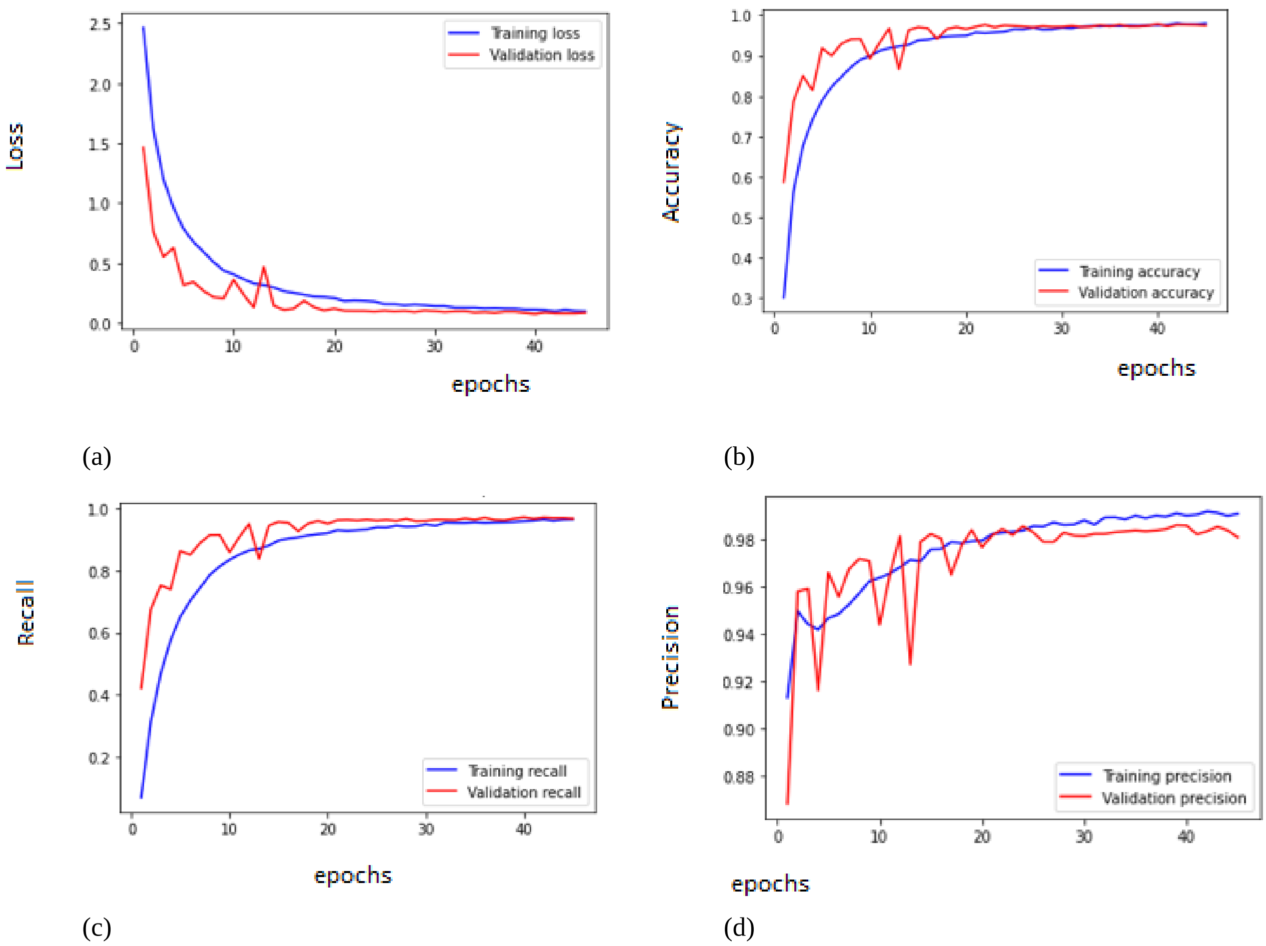


Figure6. Parameters convergence plot obtained using SGD optimizer with a learning rate of 0.001 (a) Training and validation loss, (b) Training and validation accuracy, (c) Training and validation recall, (d) Training and validation precision

4.3.3 Recognition results achieved by usingRMSprop Optimizer

Here, optimization is done using RMSprop optimizer. Parameters convergence plot for the same is shown in Figure 7. In Figure 7(a), number of spikes are present for validation loss till 25th epoch and later started decreasing for each epoch. From the Figure 7(b), 7(c) and 7(d), it can be observed that the validation accuracy, recall and precision are almost linearly varying after 25th epoch but many fluctuations are present in the values before that.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

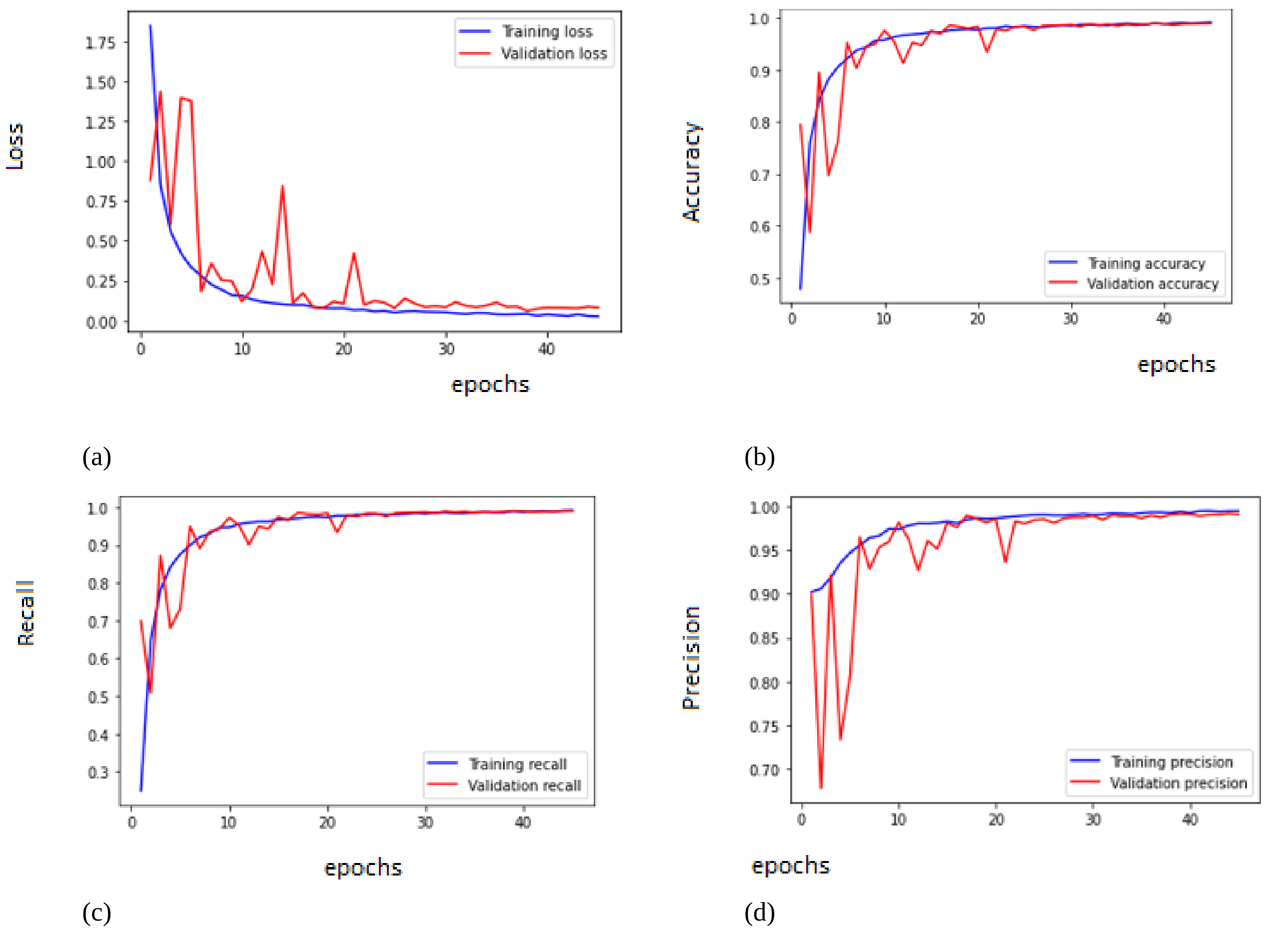


Figure7. Parameters convergence plot obtained using RMSprop optimizer with a learning rate of 0.001 (a) Training and validation loss, (b) Training and validation accuracy, (c) Training and validation recall, (d) Training and validation precision

4.3.4 Comparative Analysis of proposed method using three optimizers

In this section, various parameters like training loss, validation loss, validation accuracy, validation recall and validation precision obtained using three optimizers namely Adam, SGD and RMSprop are compared and analyzed. It can be observed from the Table 3that minimum training loss and validation loss obtained by Adam, SGD and RMS prop is (0.04, 0.07), (0.09, 0.07), (0.02,0.07). From these values, it can be observed that all the three optimizers behaved equally same for validation dataset. Maximum validation accuracy of 0.99 is obtained by Adam and RMSprop while it is less for SGD optimizer. Highest validation recall of 0.99 is given by Adam and highest precision of 0.99 is obtained by Adam and RMSpropboth. Validation Accuracy is the main parameter for any recognition model. Adam and RMSprop has performed almost same for the maximum validation accuracy which is obtained as 0.99. In terms of average validation accuracy Adam has performed better as compared to SGD and RMSprop. Comparative analysis of validation accuracy obtained by all the three optimizers corresponding to each epoch is also shown in Figure 8, from where it can be concluded that Adam optimizer has given higher validation accuracies for most of the epochs as compared to SGD and RMSprop.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

Table 3. Comparison table for three different optimizers

Optimizer	Epoch	Training loss	Validation loss	Validation Accuracy	Validation Recall	Validation Precision
Adam	1	2.63	0.86	0.77	0.64	0.89
	2	1.18	1.03	0.74	0.67	0.84
	3	0.79	2.33	0.60	0.55	0.68
	4	0.56	0.69	0.89	0.8	0.84
	5	0.44	0.19	0.95	0.94	0.96
	6	0.39	0.69	0.91	0.9	0.93
	7	0.3	0.18	0.96	0.95	0.96
	8	0.25	0.15	0.97	0.96	0.97
	9	0.22	0.29	0.97	0.96	0.97
	10	0.23	0.21	0.96	0.95	0.96
	35	0.05	0.13	0.98	0.98	0.98
	36	0.05	0.09	0.98	0.98	0.98
	37	0.06	0.14	0.98	0.98	0.98
	38	0.05	0.26	0.98	0.98	0.98
	39	0.05	0.14	0.98	0.98	0.98
	40	0.04	0.12	0.98	0.98	0.98
	41	0.05	0.4	0.97	0.96	0.97
	42	0.04	0.09	0.98	0.98	0.98
	43	0.05	0.34	0.98	0.98	0.98
	44	0.04	0.07	0.98	0.98	0.99
	45	0.04	0.09	0.99	0.99	0.99
	Average values obtained	**0.2**	**0.28**	**0.95(95%)**	**0.94**	**0.96**
SGD	1	2.6	1.46	0.58	0.41	0.86
	2	1.74	0.76	0.78	0.67	0.95
	3	1.24	0.55	0.84	0.75	0.95
	4	1.00	0.62	0.81	0.73	0.91
	5	0.81	0.31	0.91	0.86	0.96
	6	0.68	0.34	0.89	0.85	0.95
	7	0.61	0.26	0.92	0.88	0.96
	8	0.51	0.21	0.94	0.91	0.97
	9	0.43	0.20	0.94	0.91	0.97
	10	0.40	0.36	0.89	0.86	0.94
	35	0.13	0.09	0.97	0.96	0.98
	36	0.12	0.08	0.97	0.97	0.98
	37	0.12	0.09	0.97	0.96	0.98

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

	38	0.12	0.09	0.97	0.96	0.98
	39	0.11	0.08	0.97	0.96	0.98
	40	0.11	0.07	0.97	0.97	0.98
	41	0.10	0.08	0.97	0.96	0.98
	42	0.09	0.08	0.97	0.97	0.98
	43	0.11	0.08	0.97	0.97	0.98
	44	0.09	0.08	0.97	0.97	0.98
	45	0.09	0.08	0.97	0.96	0.98
	Average values obtained	**0.35**	**0.20**	**0.94(94%)**	**0.90**	**0.94**
RMSprop	1	2.43	0.88	0.79	0.69	0.89
	2	0.96	1.43	0.58	0.50	0.67
	3	0.60	0.60	0.89	0.87	0.92
	4	0.42	0.69	0.69	0.67	0.73
	5	0.34	1.37	0.76	0.73	0.80
	6	0.28	0.18	0.95	0.94	0.96
	7	0.22	0.35	0.90	0.89	0.92
	8	0.20	0.25	0.94	0.93	0.95
	9	0.15	0.24	0.95	0.94	0.95
	10	0.15	0.12	0.97	0.97	098
	35	0.04	0.11	0.98	0.98	0.98
	36	0.04	0.08	0.98	0.98	0.98
	37	0.03	0.08	0.98	0.98	0.98
	38	0.03	0.06	0.98	0.98	0.99
	39	0.03	0.07	0.99	0.98	0.99
	40	0.03	0.08	0.98	0.98	0.99
	41	0.03	0.08	0.98	0.98	0.98
	42	0.02	0.07	0.98	0.98	0.99
	43	0.04	0.07	0.98	0.98	0.99
	44	0.02	0.08	0.98	0.98	0.99
	45	0.02	0.08	0.99	0.98	0.99
	Average values obtained	**0.15**	**0.26**	**0.94(94%)**	**0.92**	**0.92**

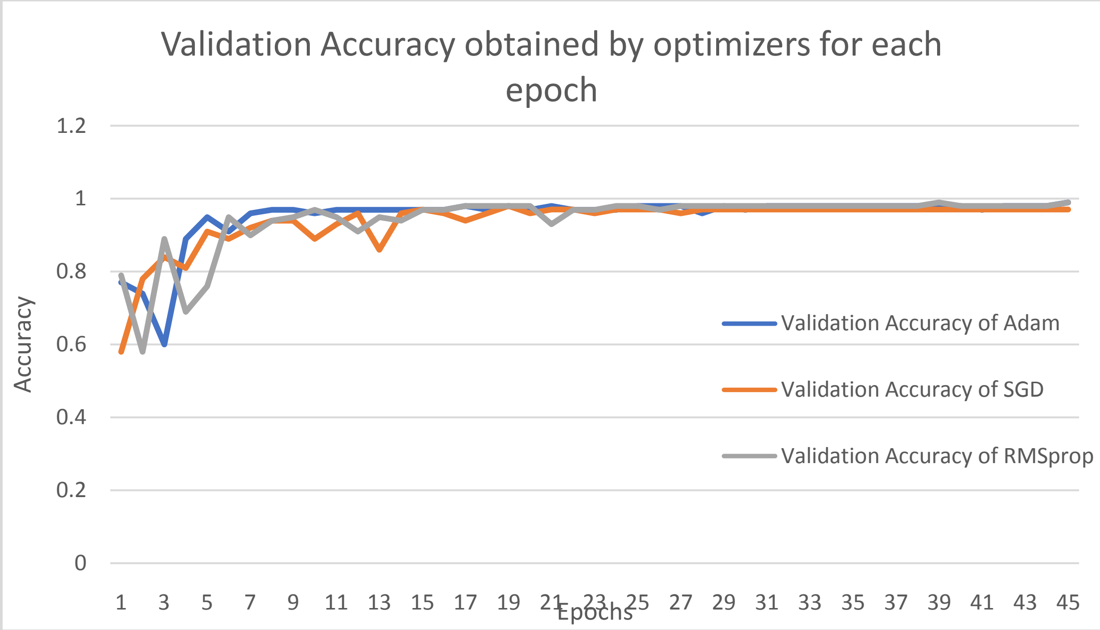


Figure8. Comparative analysis of all Adam,SGD and RMSprop in terms of validation accuracy obtained

5. CONCLUSION

Identification of the text written in Gurmukhi script is a cumbersome process due to its cursive and closely written characters. These days, deep learning is being employed many research application areas for the purpose of recognition ad classification. In this work, a 12 layered CNN model is employed for the recognition of district names of Punjab(state) which has 22000 of images. The proposed model is employed using three optimizers: Adam, SGD and RMSprop. The maximum validation accuracy obtained by Adam, SGD and RMSprop is 0.99(99%), 0.97(97%) and 0.99(99%) while average validation accuracy obtained is 0.95(95%), 0.94(94%) and 0.94(94%). It can be concluded that Adam has outperformed SGD optimizer and RMS optimizer for the recognition of district names of Punjab(State).

References

[1] Koerich AL, Sabourin R, Suen CY (2003) Large vocabulary off- line handwriting recognition: a survey. Pattern Anal Appl6 (2):97–121

[2] PlötzT,FinkGA(2009)Markovmodelsforofflinehandwritingrecognition:asurvey.IntJDocAnalRecognit12(4):269–298

[3] Madhvanath S, Govindaraju V (2001) The role of holistic para- digms in handwritten word recognition. IEEE Trans Pattern Anal Mach Intell23(2):149–164.

[4] Bhowmik, S., Malakar, S., Sarkar, R., Basu, S., Kundu, M. and Nasipuri, M., 2019. Off-line Bangla handwritten word recognition: a holistic approach. *Neural Computing and Applications*, *31*(10), (pp.5783-5798), Springer.

[5] Daniyar, N., Kairat, B., Maksat, K. and Anel, A., 2019, December. Classification of handwritten names of cities using various deep learning models.In *2019 15th International Conference on Electronics, Computer and Computation (ICECCO)* (pp. 1-4).IEEE.

Annals of R.S.C.B., ISSN:1583-6258, Vol. 25, Issue 3, 2021, Pages. 3294 - 3306
Received 16 February 2021; Accepted 08 March 2021.

[6]. U. Pal, R. K. Roy and F. Kimura: Multi-lingual City Name Recognition for Indian Postal Automation. 2012 International Conference on Frontiers in Handwriting Recognition, Bari, pp. 169-173, 2012.
[7]. Thadchanamoorthy, S., Kodikara, N.D., Premaretne, H.L., Pal, U. and Kimura, F., 2013, August. Tamil handwritten city name database development and recognition for postal automation.In *2013 12th International Conference on Document Analysis and Recognition* (pp. 793-797).IEEE.
[8]. Vajda, S., Roy, K., Pal, U., Chaudhuri, B.B. and Belaid, A., 2009. Automation of Indian postal documents written in Bangla and English. *International Journal of Pattern Recognition and Artificial Intelligence*, *23*(08), pp.1599-1632.
[9]. U. Pal, R. K. Roy and F. Kimura: Handwritten Street Name Recognition for Indian Postal Automation. 2011 International Conference on Document Analysis and Recognition (ICDAR), Beijing, pp. 483-487, 2011.
[10] K. Roy, S. Vajda, U. Pal and B. B. Chaudhuri: A system towards Indian postal automation. Ninth International Workshop on Frontiers in Handwriting Recognition, pp. 580-585. 2004
[11]. U. Pal, R. K. Roy, K. Roy and F. Kimura: Indian Multi-Script Full Pin-code String Recognition for Postal Automation. 10th International Conference on Document Analysis and Recognition (ICDAR), Barcelona,pp. 456-460, 2009.
[12] Yamashita, R., Nishio, M., Do, R.K.G. and Togashi, K., 2018. Convolutional neural networks: an overview and application in radiology. *Insights into imaging*, *9*(4), pp.611-629.
[13]. Shin, H.C., Roth, H.R., Gao, M., Lu, L., Xu, Z., Nogues, I., Yao, J., Mollura, D. and Summers, R.M., 2016. Deep convolutional neural networks for computer-aided detection: CNN architectures, dataset characteristics and transfer learning. *IEEE transactions on medical imaging*, *35*(5), pp.1285-1298.
[14]. Brownlee, J., 2016. *Deep learning with Python: develop deep learning models on Theano and TensorFlow using Keras*. Machine Learning Mastery.
[15] Zhang, Z., 2018, June. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)* (pp. 1-2).IEEE.
[16]. Yang, J. and Yang, G., 2018. Modified convolutional neural network based on dropout and the stochastic gradient descent optimizer. *Algorithms*, *11*(3), p.28.
[17] Taqi, A.M., Awad, A., Al-Azzo, F. and Milanova, M., 2018, April. The impact of multi-optimizers and data augmentation on TensorFlow convolutional neural network performance. In *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)* (pp. 140-145). IEEE.
[18] Kandel, I. and Castelli, M., 2020. The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset. *ICT express*, *6*(4), pp.312-315.