Asian Journal of Research in Computer Science

6(2): 16-24, 2020; Article no.AJRCOS.59633
ISSN: 2581-8260

Sign Language Digit Recognition Using Different Convolutional Neural Network Model

Md. Bipul Hossain[1*], Apurba Adhikary[1] and Sultana Jahan Soheli[1]

[1]Department of Information and Communication Engineering, Noakhali Science and Technology University, Noakhali − 3814, Bangladesh.

Authors' contributions

This work was carried out in collaboration among all authors. Author MBH designed the study, performed the simulation analysis, wrote the protocol and wrote the first draft of the manuscript. Author AA managed the analyses of the study. Author SJS managed the literature searches. All authors read and approved the final manuscript.

Article Information

DOI: 10.9734/AJRCOS/2020/v6i230154
Editor(s):
(1) Dr. Young Lee, Texas A&M University–Kingsville, United States of America.
Reviewers:
(1) Sarvesh Patil, Veermata Jijabai Technological Institute (VJTI), India.
(2) Shailendra Giri, Gurukula Kangri Vidyapeeth, India.
(3) J. Bethanney Janney, Sathyabama Institute of Science and Technology (SIST), India.
Complete Peer review History: http://www.sdiarticle4.com/review-history/59633

Original Research Article

Received 28 May 2020
Accepted 05 August 2020
Published 11 August 2020

ABSTRACT

An enormous number of world populations in current time are unique in that sense that they have no broad language because of the absence of their hearing capability. The people with hearing impairment have their own language called Sign Language however it is hard for understanding to general individuals [1]. Sign digits are additionally a significant piece of gesture based communication. So a machine interpreter is important to permit them to speak with general individuals. For making their language justifiable to general individual's computer vision based arrangements are notable these days. In this exploration of work we target to develop a model based on CNN to deal with the recognition of Sign Language digits. A dataset of 10 classes is used to train (70%), validation (20%) and test (10%) of the network. We consider three different models of CNN network to train and test the accuracy of sign digit. Among the three model transfer learning based pre-trained CNN performs better with test accuracy of 92%.

Keywords: Sign language; convolutional neural network; transfer learning; digit recognition.

**Corresponding author: E-mail: bipulhossainiceiu@gmail.com;*

1. INTRODUCTION

The method of human communication, either spoken, composed or representative, comprising of the utilization of words or signs in an organized and traditional manner is the language. Individuals use language to speak with one another. In our society, a few people are shockingly impeded because of the restrictions on their talking and listening abilities [2]. As they are restricted in utilizing dialects, they need to utilize uncommon signs to convey and to express their sentiments. Sign language is the essential method of imparting among tuning in and hearing impeded individuals utilizing hand and emblematic motion rather than sound or communication in language, while the general individuals don't utilize it [3]. It is extremely difficult for the general individuals to speak with the talking and listening impeded individuals on the grounds that the communication via gestures isn't reasonable for them. Consequently, it is fundamental to available a helper function which converts the communication via gestures to the general language. To partially mitigate this issue in this paper we utilize three different classification models to recognize hand sign digit. Sign Language (SL) recognition by computer vision is a challenging task due to the complexity in SL signs, large intraclass variations, and constant occlusions [4]. Over the most recent couple of years, it is profound that convolutional neural network (CNN) has demonstrated the extraordinary exhibition in the field of image classification, machine learning, pattern recognition, human activity analysis, object recognition, segmentation ,image super resolution, object detection, tracking, scene understanding, and image captioning [5]. One significant preferred position of the convolutional neural system is that the training of CNN doesn't rely upon manual feature extraction. It extracts itself high intensity features with more details and gives regularly preferable execution over customary shallow neural system [6]. In this paper we utilize three distinctive model of CNN architecture as appeared in Figs. 3, 4 and 5 and we named those as SCNN, MCNN and TCNN. The general model of CNN with only 2 hidden layers is named as SCNN. Network with progressively concealed layers are nevertheless not very high is named as MCNN and system which utilize transfer learning i.e. a pre trained network (Inception V3) is named as TCNN. The original Inception V3 was trained using a dataset of 1,000 classes from the original imagenet dataset which was trained with over 1 million training images. A dataset of 2062 image of 10 classes hand sign digit has collected form kaggle.com is utilized to train and test each model. In D. Deora et al. (2012) Indian Sign Language Numbers have been effectively recognized. A framework for an HCI able to recognize signs from Indian sign language with PCA (Principal Component Analysis) is represented by them in this paper [7]. In A. Agarwal et al. (2013) computer vision algorithms is used and they construct a characteristics depth and motion profile for sign language digits [8]. The generated feature matrix was then trained with SVM classifier. Pigou B. et al. (2015) introduce a recognition system using convolutional neural networks (CNNs), the Microsoft Kinect, and GPU acceleration and making complex handcrafted features [9]. In Oyewole et al. (2018) a Yoruba Sign Language recognition system [10] using Artificial Neural Networks (ANN) and image processing has been introduced. The association of rest of the paper is as per the following; in section 2, the hypothetical foundations are talked about in detail. Experimental details are described in section 3. The results and discussion is discussed in section 4 and section 5 presents the conclusion of the paper.

2. THEORITICAL BACKGROUND

2.1 Convolutional Neural Network

Over the past decade in a variety of fields related to pattern recognition; from image processing to voice recognition Convolutional Neural Network has had ground breaking results. There are four main operations in the CNN are Convolution, Non Linearity (ReLU), Pooling or Sub Sampling and Classifications (Fully Connected Layer) [11].

2.1.1 Convolution layer

Convolutional layer likewise alluded to as Conv. Layer is the basis of the CNN. It carries out the center tasks of training and therefore firing the neurons of the network. It performs the convolution operation over the input volume. The convolution for one pixel in the following layer is determined by the formula (1) [11]. Where N(t,f) is the output of the next layer, x is the input image and w is the kernel or filter matrix and * is the convolution operation.

$$N(t,f) = (x*w)[t, f] = \sum_m \sum_n x[m, n]\, w[t - m, f - n] \quad (1)$$

2.1.2 Pooling

The principle thought of pooling is down-sampling so as to reduce the complexity for further layers. It can be considered as similar to reducing the resolution in the image processing domain. Pooling does not affect the number of filters. Max-pooling is one of the most widely recognized kinds of pooling techniques. It partitions the image to sub-region rectangles and it only returns the maximum value inside of that sub-region. One of the most well-known filter sizes utilized in max-pooling is 2×2.[11,6]

2.1.3 Fully connected layer

The Fully connected layer is a conventional Multi-Layer perceptron that utilizes a softmax activation function at the output layer .The expression “Fully Connected” implies that each neuron in the past layer is associated with each neuron on the following layer .The output from the convolutional and pooling layers represent high-level features of the input image. The motivation behind the Fully Connected layer is to utilize these features for classifying the input image into different classes dependent on the training dataset. Apart from classification, adding a fully-connected layer is also a (usually) cheap way of learning non-linear combinations of these features. The greater part of the features from convolutional and pooling layers might be useful for the classification task, yet mixes of those features might be even better. [11,6]

2.2 Transfer Learning

Transfer learning can be define as, given a source area D_S and learning task T_S, an objective space D_T and learning task T_T, transfer learning aims to help improve the learning of the objective prescient capacity $f_T(\bullet)$ in D_T utilizing the information in D_S and T_S, where D_S in no equal to D_T or T_S isn't equal to T_T [12,13]. Transfer learning is a significant tool in AI to take care of the essential issue of deficient training data. It attempts to move the information from the source space to the objective area by loosening up the suspicion that the training data and the test information must be identical. This will prompts an extraordinary constructive outcome on numerous spaces that are hard to improve as a result of inadequate preparing data [12]. In deep learning, transfer learning is a method whereby a neural network model is first trained on a problem alike to the problem that is being solved. One or more layers from the trained model are then utilized in a new model trained on the problem of interest. Transfer learning has the advantage of diminishing the training time for a neural network model and can bring about lower speculation mistake. The weights in re-utilized layers may be utilized as the beginning stage for the training process and adapted in response to the new issue which treats transfer learning as a type of weight initialization method. This might be helpful when the primary related issue has much more labeled data than the problem of interest and the likeness in the structure of the problem may be valuable in both contexts. Three more popular models for transfer learning are i) VGG (e.g. VGG16 or VGG19) ii) Google Net (e.g. InceptionV3) iii) Residual Network (e.g. ResNet50) [14]. These models are generally utilized for transfer learning in view of their performance, yet in addition since they were models that presented explicit compositional developments, namely consistent and repeating structures (VGG), inception modules (GoogLeNet), and residual modules (ResNet) [12].

3. EXPERIMENTAL DETAILS

3.1 Dataset Descriptions

We use a dataset Sign Language Digit Dataset (SLDD) of hand sign digit collected from kaggle.com [15] which contain total of 2062 images of 10 class .Each of the class contains RGB image of hand sign digit. We divide this dataset 70% for training, 20% for validation and 10% for test. Example image (gray version) of each class is shown in Fig. 1. Four different hand image of each class with the name of the sign at top is given at the figure.

3.2 Network Architecture

For recognizing hand sign digit three different architecture which comprises of a few convolutional layers, pooling layers, normalization layers, and dense or fully connected layers are used. For simplifying the architecture drawing several layers are combined within a block as demonstrated Fig. 2. In the first model of SCNN as appeared in Fig. 3 a straightforward CNN model with just 2 layers of convolution and maxpooling followed by a flatten layers and dense layers are used for classification. Input shape (100,100,3) and 32 filters of size 3×3 is used at the first convolutional layer after that a maxpooling layer with filter size 2×2 is utilized for the decrease of information without harming the significant

features. The output of this block is then goes into a second convolutional layer, which has 64 filters and the size of filters is 3×3. After completing the second convolutional yield is feed into max pooling layer, which has the filter size of 2×2. Finally the output of the second block is provided to the flatten layer after that the two fully connected layer for final classification. Number of classes used in the final layer is 10.

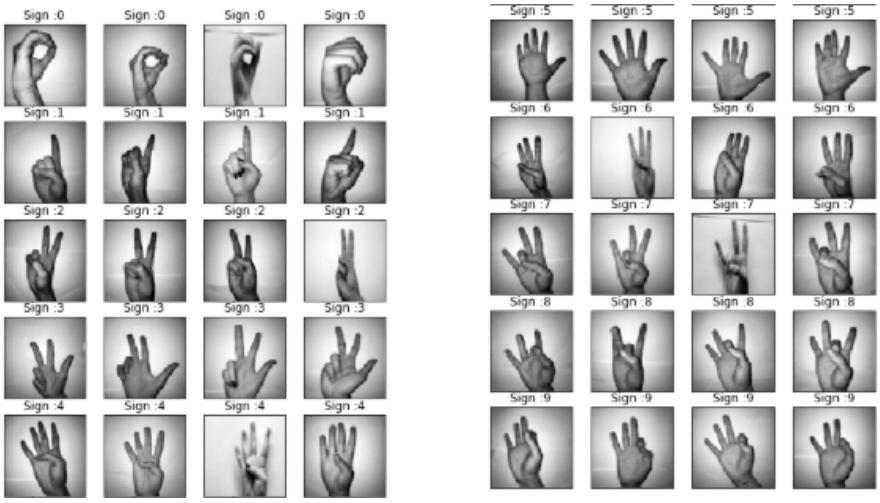

Fig. 1. Example image data of hand sign digit dataset [15]

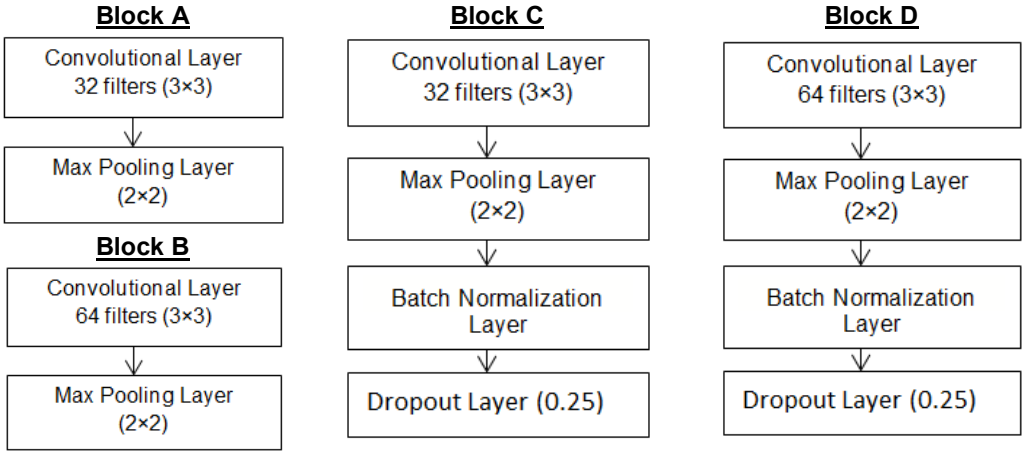

Fig. 2. Blocks of the Architecture

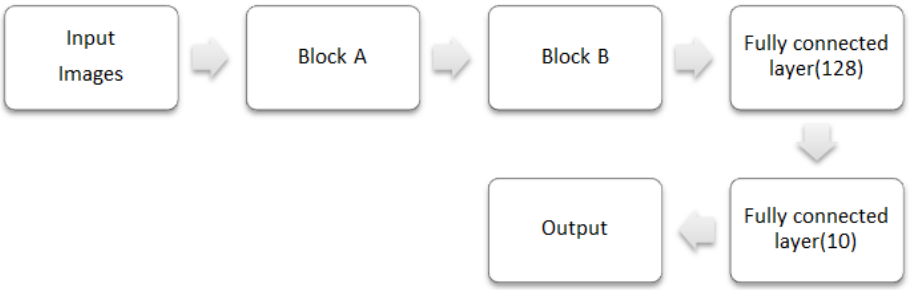

Fig. 3. Network architecture of SCNN model

The architecture of the second model as we notice already as MCNN is appeared in Fig. 4 comprise of a solitary Block C and three consecutive Block D and after that two fully connected layers. Due to the quantity of hidden layer of this model is more than the past one it performs more hidden operations than the previous model. Block C consists of a convolutional layer after that maxpooling layer then batch normalization layer and lastly a droupout layer. Block D is contrast from block C in that it utilizes the number of filter is 64 instead of 32. In the third model TCNN as shown in Fig. 5, transfer learning is used to train the network. One of the most well-known pre trained model called inception V3 is used for the training process. At the top of the pre-trained model two fully connected layer is used for sign digit recognition and a droupout layer also used for avoiding overfitting. The concatenation of this architecture's model summary is shown in Table 1. Except the end every layer used relu as activation function. The equation which define relu activation is

$$R(\alpha) = \max(0,\alpha) \quad (2)$$

At the end of every network softmax is used as activation function which calculates the probability of each class.

4. RESULTS AND DISCUSSION

Transfer learning becoming popular day by day due to it's facility of less time required for training and it also perform better for small data sets. In this study three different model of CNN are considered for recognition of hand sign digits. Total of 2062 images of 10 classes is applied with 70% for training, 20% for validation and 10% for test to each of the above mention CNN network model. The experimental results graph of training vs. validation accuracy is given in the Fig. 6 for TCNN, Fig. 7 for MCNN and Fig. 8 for SCNN. A comparative result of training accuracy and validation accuracy is given in Table 2. According to the trial results as the quantity of hidden layer is increasing in the network the accuracy of the model is also increasing but a large number of hidden layer needs a huge number of time for the training of the network. Hence a pre trained network like inception v3 is used to train the network.

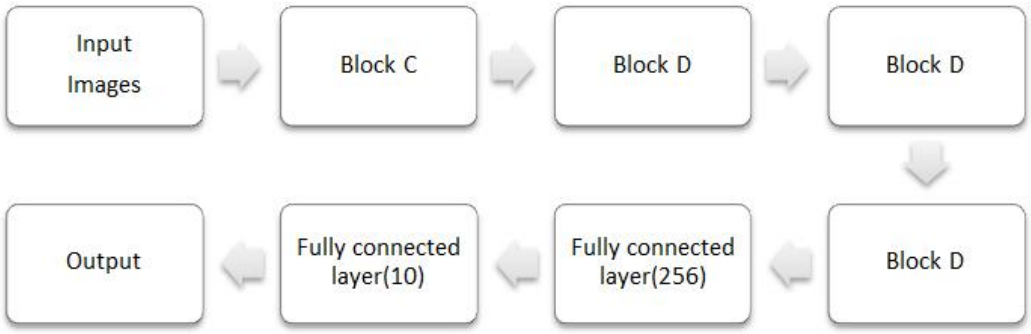

Fig. 4. Network architecture of MCNN model

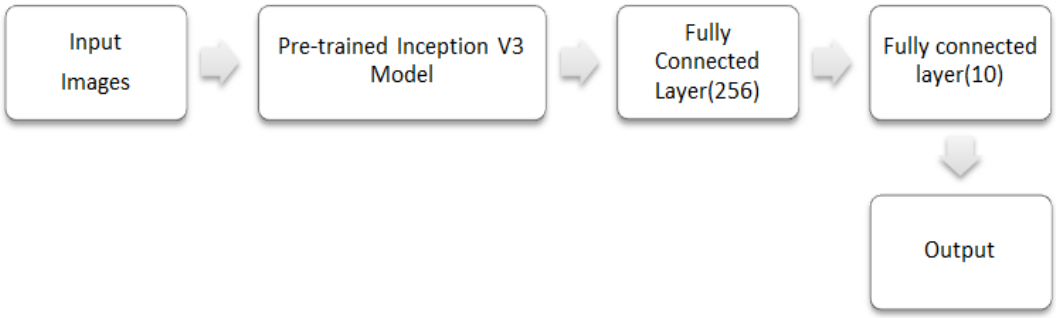

Fig. 5. Network architecture of TCNN model

Table 1. Concatenation of fully connected layer with Inception V3 mixed7

mixed7 (Concatenate)	(None, 4, 4, 768)	0	activation_60[0][0] activation_63[0][0] activation_68[0][0] activation_69[0][0]
flatten (Flatten)	(None, 12288)	0	mixed7[0][0]
dense (Dense)	(None, 256)	3145984	flatten[0][0]
dropout (Dropout)	(None, 256)	0	dense[0][0]
dense_1 (Dense)	(None, 10)	2570	dropout[0][0]

Total params: 12,123,818
Trainable params: 3,148,554
Non-trainable params: 8,975,264

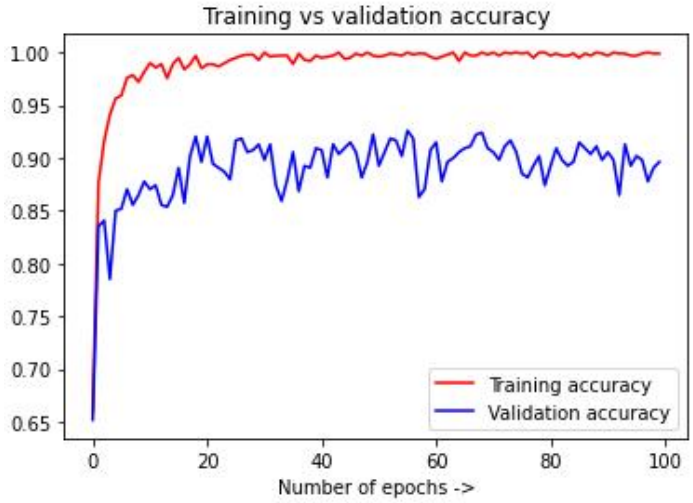

Fig. 6. Training and validation accuracy of TCNN model for digit recognition

After 100 epochs of the training process three models SCNN, MCNN and TCNN provides validation accuracy of 0.72, 0.7667 and 0.92 respectively. Validation loss we get after 100 epochs are 3.7224, 1.4030 and 0.79955 for SCNN, MCNN and TCNN.We also notice at the simulation graph of the SCNN model is approximately becoming stable after 40 epoch, on the other hand other two model is not finally stable even after 100 epochs but provides more accuracy. Since our main concern is to increase the validation accuracy, so the model which provides more accuracy is our promising model. Moreover in is experiment the validation accuracy is not too high due to the volume of training and testing data. Fig. 9 shows the confusion matrix where rows represent the true label and columns correspond to predicated label of the digit. Where true label means actual digits and predicted label means what the model predict for a digit in recognition task. For example digit 1, it matches 19 times with true label and predicted label and doesn't match only for a single case. So digit 1 is 19 times predicted as 1 and single times as 3. The number of samples we take for test is 20 per class. The classifier recognizes all samples of each class with around 95% accuracy. The comparative graph of training and validation accuracy of three different models is shown in Fig. 10.

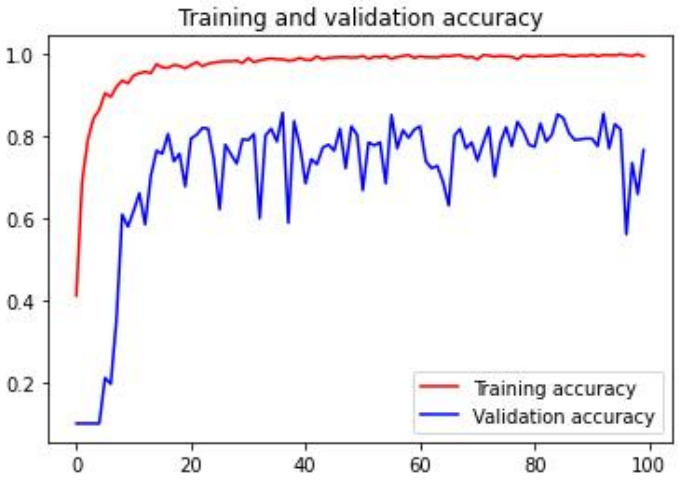

Fig. 7. Training and validation accuracy of MCNN model for digit recognition

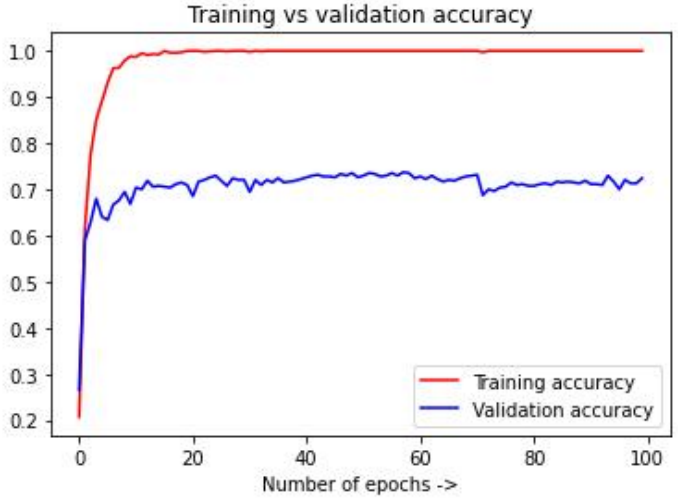

Fig. 8. Training and validation accuracy of SCNN model for digit recognition

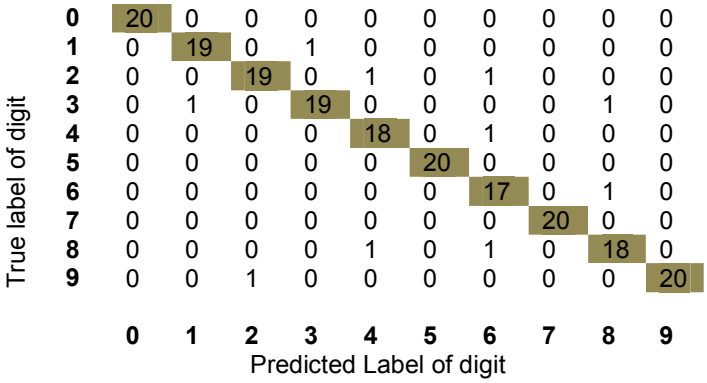

Fig. 9. Confusion matrix of the TCNN classification performance on hand sign language digit test data

Table 2. Comparative results of different model

Model	Epochs	Training accuracy	Validation accuracy	Training loss	Validation loss	Elapsed time per step
SCNN	100	0.999	0.72	1.7881e-09	3.7224	376ms/step
MCNN	100	0.9949	0.7667	0.0205	1.4030	379ms/step
TCNN	100	0.996	0.92	0.0257	0.7955	350ms/step

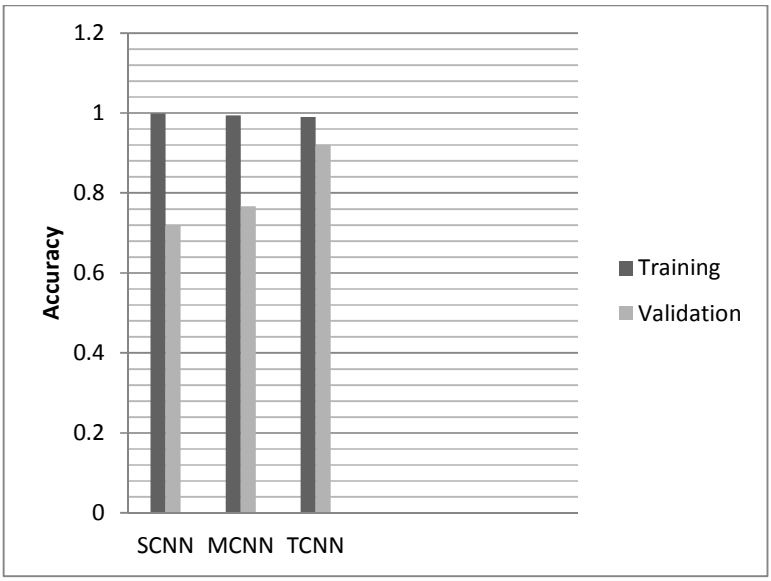

Fig. 10. Validation accuracy comparisons graph of three model

5. CONCLUSION

Deep convolutional neural network performs out breaking with respect to all other classification and recognition model. A lot of different CNN model are available in literature for accomplishing this task. Automatic feature extraction from the training images makes it's prosperity at peak. A novel hand sign digit recognition model is one of the most fundamental for hearing hindered individuals to speak with general individuals. So in this experiment we observe hand sign digit recognition with various CNN model and from the experimental results we conclude that CNN with pre trained network perform better with 92% validation accuracy. According to the confusion matrix we can also conclude TCNN model performs better with test accuracy of 95%. Despite the fact that as the number of hidden layer increases the model performances also increases but it requires lot of training time and enormous dataset. For this reason we prefer a pre trained network. So for dataset as we use in this experiment transfer learning based model is a promising methodology.

COMPETING INTERESTS

Authors have declared that no competing interests exist.

ACKNOWLEDGEMENTS

Dept. of ICE, NSTU provides all kind of logistics and others support for doing this research work. We are very much thankful to them.

REFERENCES

1. Yohanssen Pratama, et. al. Deep convolutional neural network for hand sign language recognition using model E. Bulletin of Electrical Engineering and Informatics. 2020;9(5):1873-1881. DOI: 10.11591/eei.v9i5.2027
2. Er-Rady A, Faizi R, Thami ROH, Housni H. Automatic sign language recognition: A survey. International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Fez. 2017;1-7.

DOI: 10.1109/ATSIP.2017.8075561

3. Rao GA, Syamala K, Kishore PVV, Sastry ASCS, Deep convolutional neural networks for sign language recognition. Conference on signal processing and communication engineering systems (spaces), Vijayawada. 2018;194-197.
DOI: 10.1109/spaces.2018.8316344
4. Wenjin Tao, et. al. American sign language alphabet recognition using convolutional neural networks with multiview augmentation and inference fusion; 2018.
DOI : 10.1016/J.ENGAPPAI.2018.09.006
5. Hossain MB, Naznin F, Joarder YA. Zahidul Islam M, Uddin MJ, Recognition and solution for handwritten equation using convolutionalneural network. Joint 7th international conference on informatics, electronics & vision (ICIEV) and 2018 2nd international conference on imaging, vision & pattern recognition (icIVPR), Kitakyushu, Japan. 2018;250-255.
DOI: 10.1109/ICIEV.2018.8640991
6. Stutz D, Beyer L, Understanding convolutional neural networks; 2014.
7. Deora D, Bajaj N, Indian sign language recognition. 2012 1st International conference on emerging technology trends in electronics, communication & networking, Surat, Gujarat, India. 2012;1-5.
8. Agarwal A, Thakur MK, Sign language recognition using Microsoft Kinect. Sixth International Conference on Contemporary Computing (IC3), Noida. 2013;181-185.
9. Pigou L, Dieleman S, Kindermans PJ, Schrauwen B. Sign language recognitionusing convolutional neural networks. In: Agapito L, Bronstein MM, Rother C. (eds.) ECCV 2014. LNCS. Springer, Cham. 20158925:572–578.
DOI: 10.1007/978-3-319-16178-540
10. Oyewole, Ogunsanwo Gbenga, et al. Bridging communication gap among people with hearing impairment: An application of image processing and artificial neural network. International Journal of Information and Communication Sciences. 20183(1):11.
11. Albawi S, Mohammed TA, Al-Zawi S, Understanding of a convolutional neural network. International Conference on Engineering and Technology (ICET), Antalya. 2017;1-6.
DOI:10.1109/ICEngTechnol.2017.8308186
12. Chuanqi Tan, Fuchun Sun, et. al. A survey on deep transfer learning. The 27th International Conference on Artificial Neural Networks (ICANN). arXiv: 1808.01974; 2018.
13. Pan SJ, Yang Q. A survey on transfer learning. in IEEE Transactions on Knowledge and Data Engineering. 2010; 22(10)1345-1359.
DOI: 10.1109/TKDE.2009.191.
14. Fahim Sikder M. Bangla handwritten digit recognition and generation. In: uddin m., bansal j. (eds) proceedings of international joint conference on computational intelligence. Algorithms for intelligent systems. Springer, Singapore; 2020.
15. Available:https://www.kaggle.com/ardamavi/sign-language-digits-dataset
[Accessed date:01-05-20]

Peer-review history:
The peer review history for this paper can be accessed here:
http://www.sdiarticle4.com/review-history/59633