Comparison of Different Pre-Trained Neural Networks for Gesture Recognition

N. D. Madanayaka[1], L. S. K. Udugama [1,2*], J.S. Cole[1]

Sri Lanka Technological Campus, Padukka, Sri Lanka[1], Open University, Nawala, Sri Lanka[2]

[*] udugama@ou.ac.lk

Abstract - Gesture recognition is a complex task for computers. However, researchers have developed neural networks and techniques that can be used to recognize gestures with high accuracy. In this study, two models were built, using ResNet 3D-18 and ResNet MC-18 and applying transfer learning with a subset of the 20BN Jester dataset to recognize human gestures. Consequently, higher accuracy for gesture recognition was achieved using the smaller dataset and fewer resources. Finally, the generated results are compared with the published approaches.

Keywords: Deep Learning, 3D CNN, Transfer Learning, HDF5, ResNet 3D-18, ResNet MC-18, 20BN Jester dataset

I. INTRODUCTION

Deep learning is a subset of machine learning, and its functionality is similar to the human brain. To achieve this, it contains heavy mathematical tasks which consume more resources. Further, it needs a large dataset to achieve higher accuracy. Though parallel processing can reduce computation time, training deep learning networks with large datasets and deeper neural networks may take enormous time. However, to reduce the training time, the transfer learning (TL) approach has been used [1]. It reuses already trained networks for similar tasks. Transfer learning is used in computer vision and natural language processing applications as they require big datasets for training and consume a lot of resources and time. Human Gesture recognition is a computer vision, time series task. According to [1], TL can be efficiently used for time series classifications. In this paper, we tested two neural network models using TL with a subset of a larger dataset and compared them with already published neural networks.

II. METHOD

There are widely available neural networks such as AlexNet, VGGNet, ResNet, Inception, and GoogLeNet. Out of those, ResNet 3D-18 and ResNet MC-18 have been selected and trained, using TL. For the training, the 20BN Jester dataset is chosen [2] as some of the neural networks have been already tested with it. However, only a subset of the dataset was used for our approach. Finally, the generated models have been compared with the neural networks, Inception [3] and ResNet-101 [4], which used the whole dataset.

A. Selection of the Dataset

The 20BN Jester dataset contains 148,092 short videos of basic human hand gestures of 27 classes. As for the classes, this dataset has hand gestures such as zooming out with two fingers, swiping left, shaking a hand, and so forth. The training set contains 118,562 samples. Validation and test datasets contain 14,787 and 14,743 video samples, respectively.

We used a subset of the 20BN Jester dataset, which is already available on the Kaggle platform [3]. This dataset was created by extracting the videos that have 37 frames. It has 50,420 video samples in the training set and 7047 in the validation set. It also has the same 27 classes as the original. However, the size of this dataset is about 42% of the original one.

B. Preparation of the Dataset

Initially, it was attempted to train the neural network with the dataset given in [5] and observed that it took a considerable time for training. Further, the GPU utilization was low and did not exceed 24%. After some research, it was found that reading JPG frames and decoding them took a significant time. On account of that, GPUs had to remain inactive until frames were decoded. To decrease the decoding time of the frames, we created the HDF5 [6] (Hierarchical Data Format 5) dataset. Accordingly, JPG frames were stored as uncompressed bytes in the HDF5 dataset. Moreover, we removed the first and last five frames from every video sample as most of them do not contain gestures. Consequently, every video sample has only 27 frames in the modified HDF5 dataset.

C. Selection of the Neural Networks

For the neural network, the pre-trained 3D networks were selected. 3D CNN (Convolutional Neural Network) is a three-dimensional approach to CNN. Widely used 2D CNN uses 2D input and 2D filters. Because of that 2D input matrix is multiplied with a 2D filter to do 2D convolution. But in 3D CNN, it used multiple pairs of 2D CNN to compute 3D CNN.

We used two pre-trained neural networks for our purpose.

1. ResNet 3D-18 [7]
2. ResNet MC-18 [7] (ResNet Mixed Convolutional-18)

These two networks were chosen for several reasons: they were trained for human action recognition and built focusing 3D CNN. Accordingly, it avoids using any recurrent networks to identify the dynamic actions. Also, these networks are simple in comparison to other networks. Therefore, it may reduce the network training time.

Also, we developed our model for future analysis. It is a network which is an extension of ResNet 3D-18. We removed the last fully connected layer and added three new fully connected layers to it. They had 256, 128 and 27 neurons in each layer, respectively. The above mentioned network is referred to as xResNet 3D-18 (Figure 1).

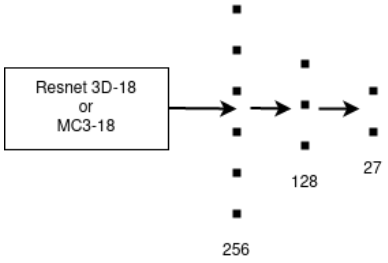

Figure 1. xResNet 3D-18

D. Training the Network

The models were experimented by adjusting the number of layers which being trained from the end. However, the same hyperparameters (Table 1) were used without changing them. Further, we used a random rotation of a maximum of 30° to add data augmentation.

No.	Parameter	Value
1	Learn Rate	1e-03
2	Optimizer	AdamW
3	Batch Size	50
4	Loss Function	Cross Entropy
5	Number of frames	27
6	Frame Size	150 * 150
7	Random Rotation	30° (max)

Table 1. Hyperparameters used

III. RESULTS

The last 11 convolutional layers and the output linear layer of the ResNet 3D-18 were trained for ten epochs. The results are given in Table 2. It achieved a training accuracy of 97.19%. Its top-1 validation accuracy is 92.91% and top-5 validation accuracy is 99.55%. After that, we tried the ResNet MC3-18 network. We trained the MC3-18 similarly but for nine epochs and got 95.41% training accuracy and 92.04% top-1 validation accuracy. Its top-5 validation accuracy is 99.43%. We used the same approach for training xResNet with different numbers of layers.

As shown in Table 2, ResNet 3D-18 and ResNet MC-18 performed better than the xResNet 3D-18. Therefore, we did not use the xResNet 3D-18 for further work. Finally, we compared our models with already published neural networks. The comparison is shown in Table 3.

IV. LIMITATIONS AND FUTURE WORK

We used the Kaggle platform for training the models. The major limitation we faced was the lack of a GPU cluster. Training video datasets with 3D CNN consumes a lot of time and resources. Therefore, we did not have enough resources to train the whole network (without transfer learning) with the 20BN Jester dataset. As for future work, we have planned to experiment with the full dataset (with 118,562 selections) and compare it with the half dataset (with 50,420 samples).

V. CONCLUSIONS

In conclusion, two models were built, using TL for pre-trained neural networks with a smaller 20BN Jester dataset. Those models were compared with the already published work that used the complete dataset. Accordingly, the results demonstrate that the proposed model performs with higher accuracy (92.91%) though the published work performs slightly better (96.9%). However, our model used only half of the dataset that was used by the published work. Further, it uses fewer resources than the other neural networks. Consequently, training networks with TL can achieve high accuracy without consuming enormous resources and time.

Network	Backbone	Training Samples	No. of Frames	Top-1 %	Top-5 %
TRG [6]	Inception	118,562	8	96.8	99.9
TRG [6]	Inception	118,562	16	96.9	99.9
MFNet-C50[7]	ResNet-101	118,562	10	96.7	99.8
Ours	**ResNet 3D-18**	**50,420**	**27**	**92.9**	**99.5**
Ours	**ResNet MC18**	**50,420**	**27**	**92.0**	**99.4**

Table 3. Validation accuracy comparison for 20BN Jester dataset

References

[1] I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.-A. Muller, “Transfer learning for time series classification,” 2018, pp. 1367–1376. doi: 10.1109/BigData.2018.8621990.

[2] J. Materzynska, G. Berger, and R. Memisevic, “The Jester Dataset: a Large-Scale Video Dataset of Human Gestures,” 2019. Accessed: Jul. 30, 2020. [Online]. Available: https://openaccess.thecvf.com/content_ICCVW_2019/papers/HANDS/Materzynska_The_Jester_Dataset_A_Large-Scale_Video_Dataset_of_Human_Gestures_ICCVW_2019_paper.pdf

[3] J. Zhang, F. Shen, X. Xu, and H. T. Shen, “Temporal reasoning graph for activity recognition,” IEEE Transactions on Image Processing, Vol. 29, Pp. 5491–5506, 2020, doi: 10.1109/TIP.2020.2985219.

[4] M. Lee, S. Lee, S. Son, G. Park, and N. Kwak, “Motion feature network: Fixed motion filter for action recognition,” in Proc. Eur. Conf. Comput. Vis. (ECCV), Pp. 387–403,Sep.2018.

[5] Danish, “20bn-jester,” www.kaggle.com, 2019. https://www.kaggle.com/datasets/toxicmender/20bn-jester (accessed Aug. 10, 2022).

[6] The HDF Group, “Hierarchical data format, version 5.”

[7] D. Tran, H. Wang, L. Torresani, J. Ray, Yann LeCun, and Manohar Paluri, “A closer look at spatiotemporal convolutions for action recognition,” CoRR, vol. abs/1711.11248, 2017, [Online]. Available: http://arxiv.org/abs/1711.11248

Table 2. Training accuracy, validation accuracy and training time comparison

Network	Epochs	No. of convolutional layers trained	No. of fully connected layers trained	Training accuracy	Top-1	Top-5	Training time	Average time per epoch (min)
Resnet 3D-18	10	11	1	97.19%	92.91%	99.5%	575 min 5 s	58
ResNet MC-18	9	11	1	95.41%	92.04%	99.4%	686 min 45 s	76
xResNet 3D-18	20	0	3	33%	-	-	494 min 49 s	25
xResNet 3D-18	10	3	3	92.57%	81.58%	-	464 min 48 s	47
xResNet 3D-18	10	5	3	95.31%	87.47%	99.0%	462 min 30 s	46
xResNet 3D-18	10	10	3	96.19%	91.95%	99.3%	536 min 57 s	54