IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.6, Special Issue.11, Dec 2018 E-ISSN: 2347-2693

Traffic Sign Recognition Using Optimized Convolutional Neural Network

[1*]Anjali T P, [2]N Ramasubramanian

[1]M.Tech. Scholar Engineering NIT, Trichy, India
[2]Dept. of Computer Science NIT, Trichy, India

Available online at: www.ijcseonline.org

Abstract— Convolutional Neural Network (CNN) is one of the most efficient Deep Neural Networks. The addition of more layers and neurons to the CNN increases its computational complexity. Even though CNNs are capable of solving many real time image recognition tasks flawlessly, it is also crucial to design optimum neural network architecture by reducing the associated memory and computational costs for resource critical applications. The proposed method optimizes a pre-trained CNN model for Traffic Sign Recognition by identifying and eliminating the redundant channels in fully connected layer of the neural network. The basis of the algorithm is that in a large neural network, the contribution of some of the neurons is negligible and can be eliminated without much effect on the overall performance. After eliminating the channels, the resulting model is retrained to compensate for the performance loss. The process of elimination and retraining is repeated until no more redundant channels are identified. The performance of the models so developed are further compared with the original model and evaluated based on accuracy and inference time. By removing 69% of the neurons in the fully connected layer, a compression rate of 2.85 was achieved and inference time got reduced by 97ms .The model so developed had accuracy slightly higher than the original model due to the retraining performed after each iteration.

Keywords *-- Convolutional Neural Network, Network Pruning, Traffic Sign Recognition*

I. INTRODUCTION

Traffic Sign Detection and its Recognition is an integral part of Advanced Driver Assistance Systems. They not only contribute to effortless driving but also ensure the safety of the driver and pedestrians. Traffic Sign recognition is carried out in two phases: the first phase is detecting the presence of traffic signs in the image and second one is to classify the image to the appropriate category. CNN is one of the most popular Deep Neural Network used in Image classification. They are known for their high recognition rate and fast execution. There has been rapid development in the field of Computer Vision in recent years leading to the development of many state of the art-neural-networks. Even though they offer high performance, deploying them in a resource constrained environment such as mobile phones or embedded gadget, is a challenge. A resource constrained environment is characterized by limited storage capacity, computational capability and power supply.

Deep Neural Networks have millions of parameters which results in heavy computational cost and storage overhead. Designing a neural network with optimal number of neurons is a tedious task. Even the most accepted neural network architectures use empirical numbers such as 128,256, 512, 4096 etc. Pruning is a popular method to compress a neural network by eliminating the redundant neurons. It is based on the fact that a significant portion of large neural network offer zero output irrespective of the data and they can be removed without much effect on the overall accuracy of the neural network model.

II. EXISTING WORK

Significant redundancy has been studied and demonstrated in several deep learning models [1] and is mainly attributed to the existence of large number of parameters in deep neural networks. The downside of having over-parameterized model is not only wastage of memory and extensive computation, but also leads to serious over fitting problem. Therefore, reducing the number of parameters is a serious problem addressed by various researchers. Most previous works on improving network architectures fall in two main categories; one concentrates on the high level architectural design and the other focuses on low level weight pruning. The introduction of average pooling layer in [2] and inception module in GoogLeNet[3] are fine examples of optimizing neural networks from high level design perspective. To reduce the number of connections and weights in neural networks, different methods have been explored. A magnitude-based approach and Hessian matrix based approach [5], to prune weights basing on numerical properties of the weights and loss functions without any external data involved. An iterative method to prune

connections in deep architectures accompanied by Quantization and Huffman coding was proposed by Han et al [6]. When all three techniques were used in pipeline, the number of parameters in the network was found to be reduced by around 10 folds.

In our method, the Average percentage of zeros [7] contributed by each neuron in Fully Connected Layer is calculated and those having this value higher than a threshold are deleted. The compressed model is then retrained to compensate for the loss incurred due to the channel deletion. This process is repeated, until no more channels are identified for deletion.

III. CONVOLUTIONAL NEURAL NETWORK (CNN)

CNN is a deep neural network used in the field of visual imagery. CNN consist of multiple hidden layers. Typically each hidden layer consists of multiple stages such as convolutional layer, pooling layer and some regularization layers such as dropout.

A. Convolutional Layer

After applying convolution operation to the input which is basically multiplication and addition of values by striding a filter across the input, the result is passed to the next layer. Although fully connected feed forward neural network can be used to learn features and classify data, this architecture is not suitable for images .This is because the process would require very high number of neurons. The Convolution operation solves this problem as it reduces the number of parameters. Hence, the network becomes weaker with less number of parameters.

B. Pooling Layer

Pooling Layer maps the output of a neuron cluster at one layer to a single neuron in the next layer. MaxPooling uses the maximum value from each cluster of neurons while Average Pooling takes the average values from all neurons in the cluster.

C. Fully Connected Layer

This layer basically connects every neuron in one layer to every neuron in the next layer. Since there are more connections in this layer, the parameter count is also large.

D. Regularization Layer

As most parameters are contributed by the fully connected layer, they are highly prone to over fitting. Dropout is a method used to handle the same. During training stage, some nodes are dropped out and a reduced network will be trained in that stage. The nodes to drop are randomly selected with a probability of 0.5. By avoiding training all the nodes, dropout helps to decrease over fitting. This also improves the training speed.

IV. DESIGN AND MODELLING

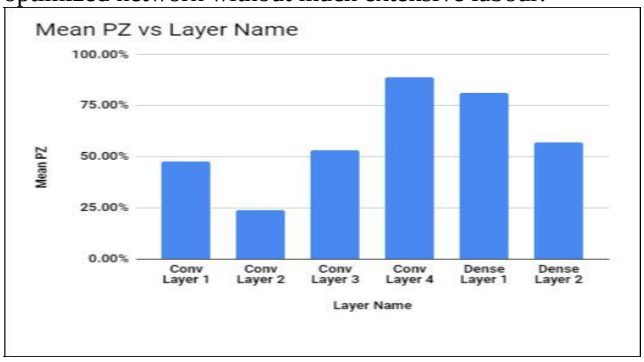
To design the model, we start from an empirically designed network for Traffic Sign Recognition; the algorithm first identifies redundant weak neurons by analyzing their activations on a validation dataset. Then those weak neurons are pruned while others are kept to initialize a new model. Finally, the new model is retrained or fine tuned depending on the performance drop. The retrained new model can maintain the same or achieve higher performance with smaller number of neurons. The model after each iteration is saved. This process can be carried out iteratively until a satisfying model is produced.

Percentage Zeros (PZ) is used to measure the percentage of zero activations of neurons. Let $O^{(i)}_c$ denotes the output of c^{th} channel in i^{th} layer,

Mean PZ of the c^{th} neuron in i^{th} layer is defined as

$$\text{Mean PZ}_c^{(i)} = \text{PZ}(O_c^{(i)}) = \frac{\sum_k^N \sum_j^M f(Oc,j^{(i)}(k)=0)}{N*M}$$

Where f *(.)*=1 if true, and *f(.)*=0 if false, *M* denotes the dimension of output feature map of $O_c^{(i)}$ and *N* denotes the total number of validation examples. The larger the number of validation examples, the more accurate is the measurement of PZ. This definition of PZ is used to evaluate the importance of each neuron in the network. Fig (1) shows the PZ calculation for the different Convolution Layers and the Fully Connected Layers(FC) in the designed model for Traffic Sign Detection. It is clear that most redundancy occurs at the higher convolution layers and fully connected layers. Since a neural network has multiplication-addition-activation computation process, a neuron which has its outputs mostly zeros will have very little contribution to the output of the subsequent layers, as well as the final output. Thus, these neurons can be removed without harming the overall accuracy of the network. In this way optimal number of neurons on each layer can found and thus obtain a optimized network without much extensive labour.



Fig(1). Graph illustrating the PZ at each layer

The pruning method consists of three steps. First, the conventional process is used to train the network and number of neurons in each layer is set empirically. The network is then run using a validation dataset to obtain the PZ of neurons belonging to each layer.

Neurons with high PZ are pruned. The connections belonging to these neurons are removed accordingly when the neuron is pruned. The trimmed network does exhibit some level of performance drop. Thus in the final step, the network is retrained to strengthen the remaining neurons to enhance the performance of the trimmed network. If the trimmed network is trained from scratch, the process is more time consuming.

To implement the network pruning, Fully Connected Layers which has highest number of parameters as well as high PZ was chosen. To decide which neurons to prune, those neurons with PZ larger than one standard deviation from the mean PZ of the target trimming layer is selected.

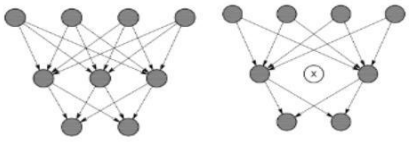


Fig (2). Neural network before and after the removal of redundant neurons

IV.IMPLEMENTATION & RESULTS

A. Experimental Setup

System used for the experimental purpose has following configuration:

Table (1).System hardware and software configurations

B. Dataset German Traffic Sign Recognition Benchmark (GTSRB)

GTSRB dataset consist of 43 different classes of traffic signs. Each image forms an array of 32*32*3, represented in RGB colour space with integer values ranging from 0 to 255.The dataset has a total of 39,209 images for performing training as well as validation. Some of the images in inference data were collected from the campus. Dataset statistics:

Table (2). Statistics of GTSRB Dataset

Training Dataset	31,367 samples
Validation Dataset	7,842 samples

Fig (3) Subset of the Traffic Sign Dataset

III. RESULTS

The initial model for Traffic sign recognition consists of 16 layers with 4 layers of convolution and 3 dense layers including output layer.

The Percentage Zeros were calculated for all 4 Convolution layers (Conv) and 2 Fully Connected (FC) Layers and is shown in Fig (1). The maximum redundancy was found to be in the FC1 and Conv Layer 3. As the parameter count is maximum in FC layer, this layer was subjected to pruning.

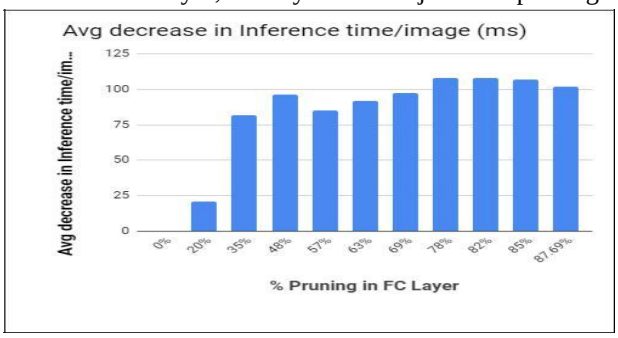


Fig (4) Graph representing the change in Inference time with Pruning in FC Layer

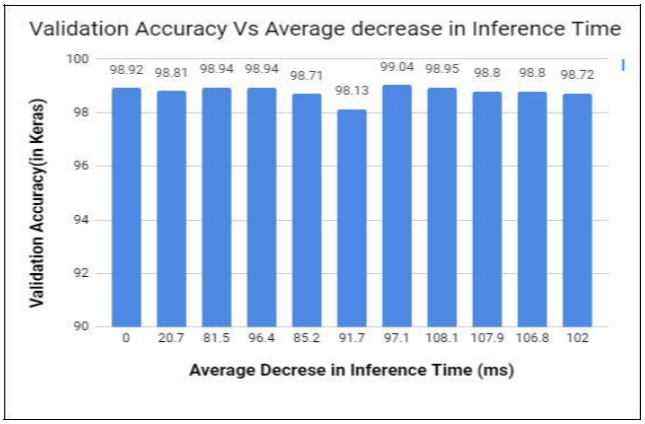


Fig (5) Graph representing the change in Accuracy with respect to Inference time

The results obtained after each stage of pruning is shown in Fig (4) and Fig (5). The inference time reduced with decrease in model size, while accuracy increased in certain cases and declined in some other. It is worth noticing that some of the pruned models showed higher accuracy than original model. This is because of the repeated training which resulted in smaller model with higher accuracy. The model seem to show acceptable performance even when more than 85% of the FC1 layer was pruned with its size reduced by 5 times.

V. CONCLUSION

Iterative Neural Network Pruning and retraining method was implemented to build an optimized model for Traffic sign detection. The Fully Connected layer (FC) and the fourth convolution layer had the maximum redundancy. Since most of the parameters were present in the fully connected layer, this layer was pruned to compress the original model. The original model was compressed by a factor of 2.85 and inference time got reduced by 97ms in our optimum model. The accuracy of the optimum model was 99.04% which is slightly higher than the actual model. This is because of the repeated training done after each pruning step. Further, the performance of each of the pruned models were studied and compared against the original model. The method can be further extended to other layers of the model to build much more compact models suitable for real time and resource critical applications.

REFERENCES

[1]. Denil, M., Shakibi, B., Dinh, L., Ranzato, M., de Freitas, N.: Predicting parameters in deep learning. CoRR abs/1306.0543(2013)

[2]. Lin, M., Chen, Q., Yan, S.: Network in network. CoRR abs/1312.4400(2013)

[3]. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S.E.,Anguelov, D., Erhan, D., Vanhoucke, V.,Rabinovich,A.: Going deeper with convolutions. CoRRabs/1409.4842(2014)

[4]. Hanson, S.J., Pratt, L.: Advances in neural information processing systems 1. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1989) 177–185

[5]. Hassibi, B., Stork, D.G.: Second order derivatives for network pruning: Optimal brain surgeon In:Advances in Neural Information Processing Systems 5, [NIPS Conference], San Francisco, CA, USA, Morgan Kaufmann Publishers Inc. (1993) 164–171

[6]. HHan, S., Mao, H., Dally, W.J.: Deep compression: Compressing deep neural networks with pruning trained quantization and Huffman coding. arXiv preprint arXiv:1510.00149 (2015)

[7]. HHengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures. arXiv preprint arXiv:1607.03250, 2016.

[8]. YYihui He ; Xiangyu Zhang ; Jian Sun : Channel Pruning for Accelerating Very Deep Neural Networks (2017), IEEE International Conference on Computer Vision (ICCV)

[9]. Kaoutar Sefrioui Boujemaa, Ismail Berrada, Afaf Bouhoute: Traffic sign recognition using convolutional neural networks, 2017 International Conference on Wireless Networks and Mobile Communications (WINCOM)

[10]. GTSRB Dataset- http://dx.doi.org/10.1016/j.neunet.2012.02.016

