Electrical Engineering Department

Design Of A BNN Inference Engine Using Quantization And Retraining Technique

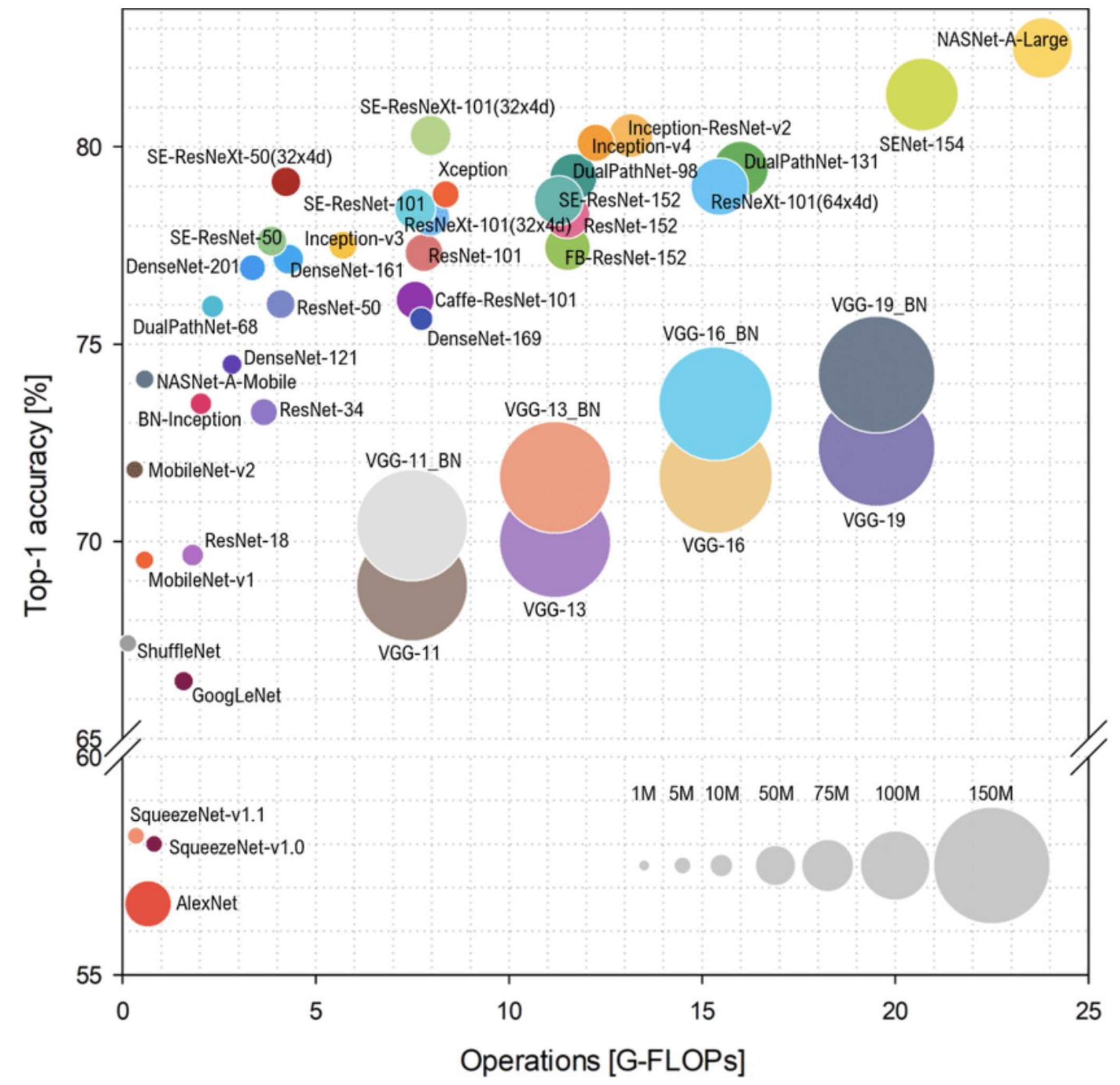
Shiva Murthy, Nivedita (MS Electrical Engineering)
Tarihalkar, Tushar (MS Electrical Engineering)

Project Advisor: Dr.Chang Choo

Introduction

BNNs are a requisite component of artificial intelligence for providing a near-human or superhuman efficiency and accuracy with reduced weights, memory occupance and power consumption.

The objective of the project was to develop a small-sized inference model i.e., a pruned network with less complexity, by quantization, net-adaptation, and retraining in order to reduce the computation cost and maintain a better efficiency and performance.

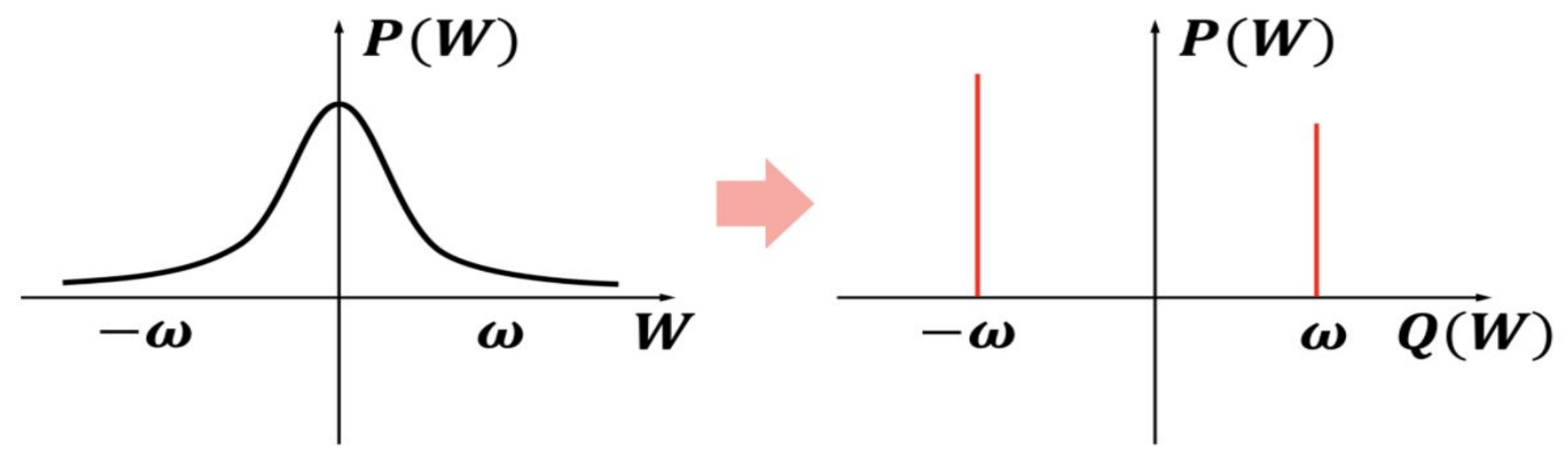


Ball Chart of Accuracy Vs Complexity

Methodology

Quantization - 1 bit weight, 8 bit feature map

- Usually a CNN consumes a huge amount of memory space which is not favourable with dense layers and increased complexity.
- A typical ResNet-101 takes around 170MB of storage space whereas in the case of AlexNet around 250MB.

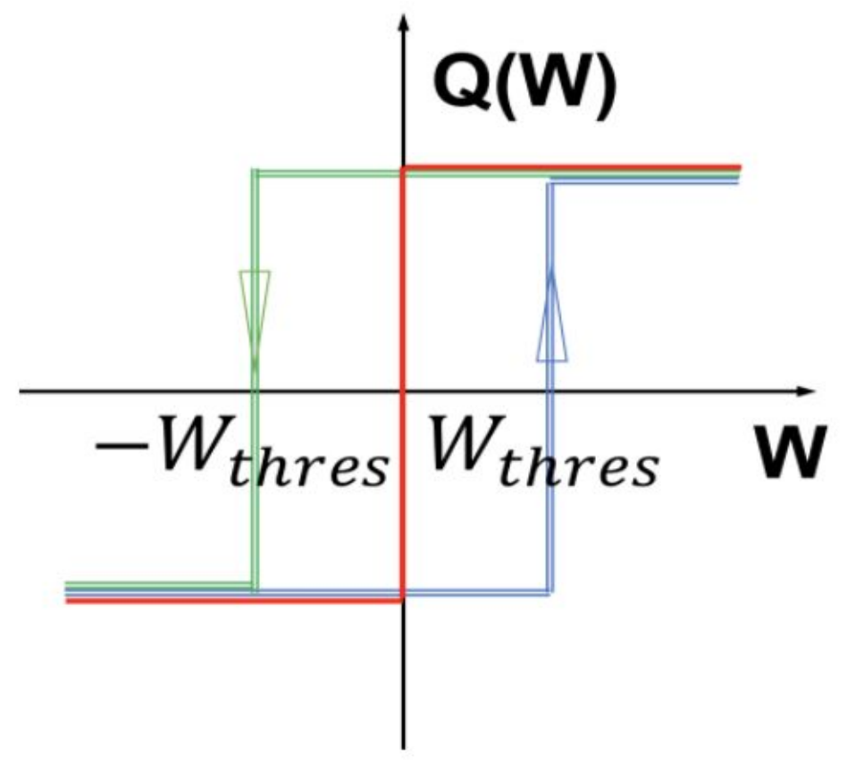


Converted 1-bit quantized weight and its hysteresis loop

- To stabilize the training process, a weight threshold of 0.1 was defined and a hysteresis loop for these 1-bit weights is illustrated in the above figure.
- Expressing the weights and activation functions in a 1-bit form ranging between (-w, +w) instead of a full-precision, thus utilizing 1-bit of memory for each variable resulting in a small binarized model of 1/32.

Methodology

- These weight optimizations are applied when the customized SGD optimizer function is called from the train class.
- Here B(x) is the binary function, with $||x||_1/n$ is the scaling factor used to maintain a value range.



$$\textbf{Forward: } x_b = B(x) = \frac{\|x\|_1}{n}\text{sign}(x),$$

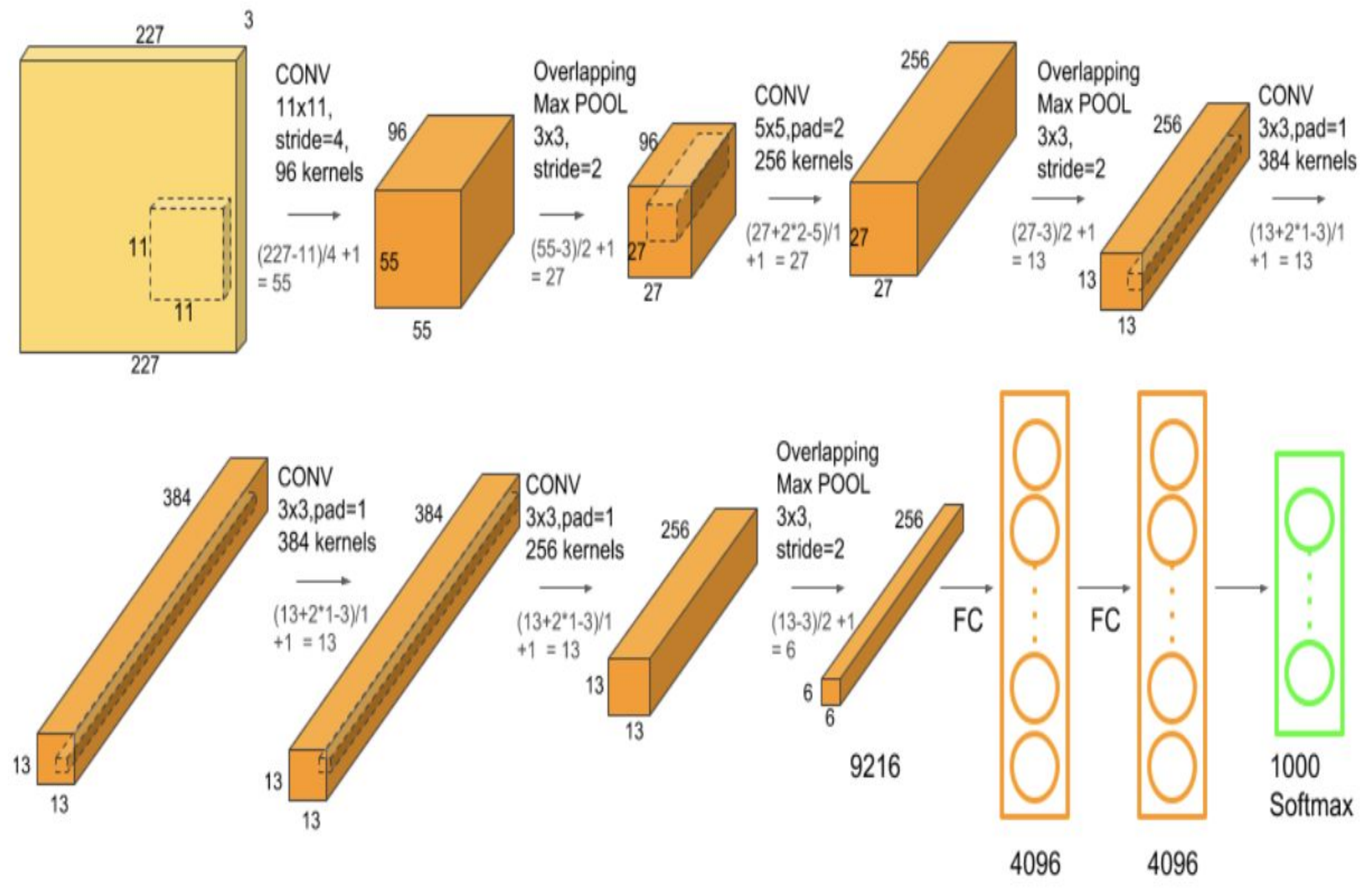
$$\textbf{Backward: } \frac{\partial B}{\partial x} \approx \mathbb{I}\{|x| < 1\}.$$

Methodology for Binarization

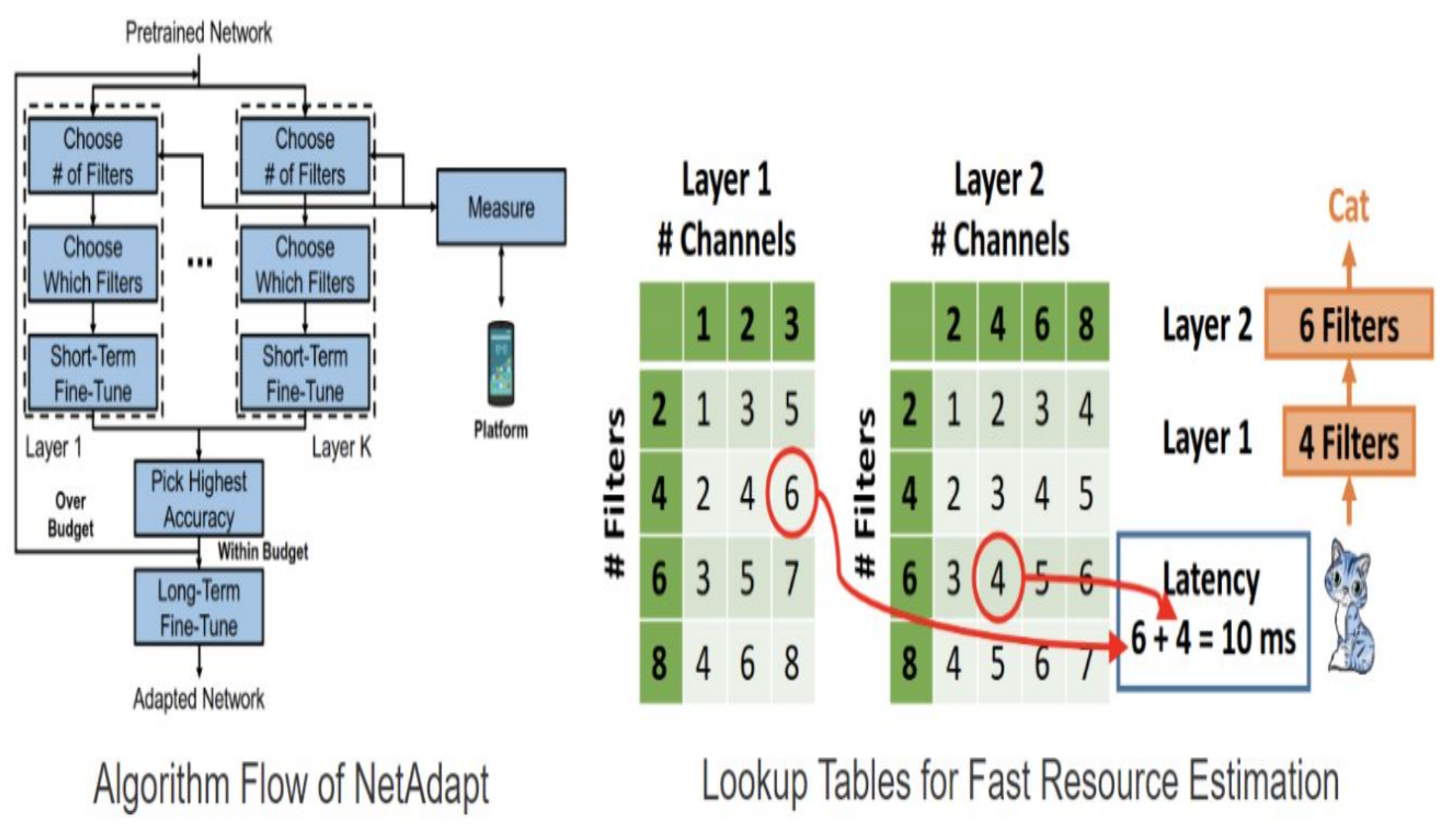
Usage of CrossEntroyp on loss function for quantization method with SGD optimizer with adjusting learning rate.

NetAdapt Applied on AlexNet

- The weights and Multiply-Accumulate Operations duplicates with the increase in the amount of resource consumed for a pre-defined network.
- To use the NetAdapt method for "Direct metrics" approach inside the optimization network. These metrics are evaluated from the empirical values obtained from the target resource.
- Most of the proposed work in order to improve the efficiency of DNNs are based on "indirect metrics".


227
3
227
11
11
CONV 11x11, stride=4, 96 kernels
(227-11)/4 +1 = 55
96
55
Overlapping Max POOL 3x3, stride=2
(55-3)/2 +1 = 27
27
CONV 5x5,pad=2 256 kernels
(27+2*2-5)/1 +1 = 27
256
Overlapping Max POOL 3x3, stride=2
(27-3)/2 +1 = 13
13
CONV 3x3,pad=1 384 kernels
(13+2*1-3)/1 +1 = 13
384
CONV 3x3,pad=1 384 kernels
(13+2*1-3)/1 +1 = 13
CONV 3x3,pad=1 256 kernels
(13+2*1-3)/1 +1 = 13
Overlapping Max POOL 3x3, stride=2
(13-3)/2 +1 = 6
6
9216
FC
FC
4096
4096
1000 Softmax

AlexNet Architecture



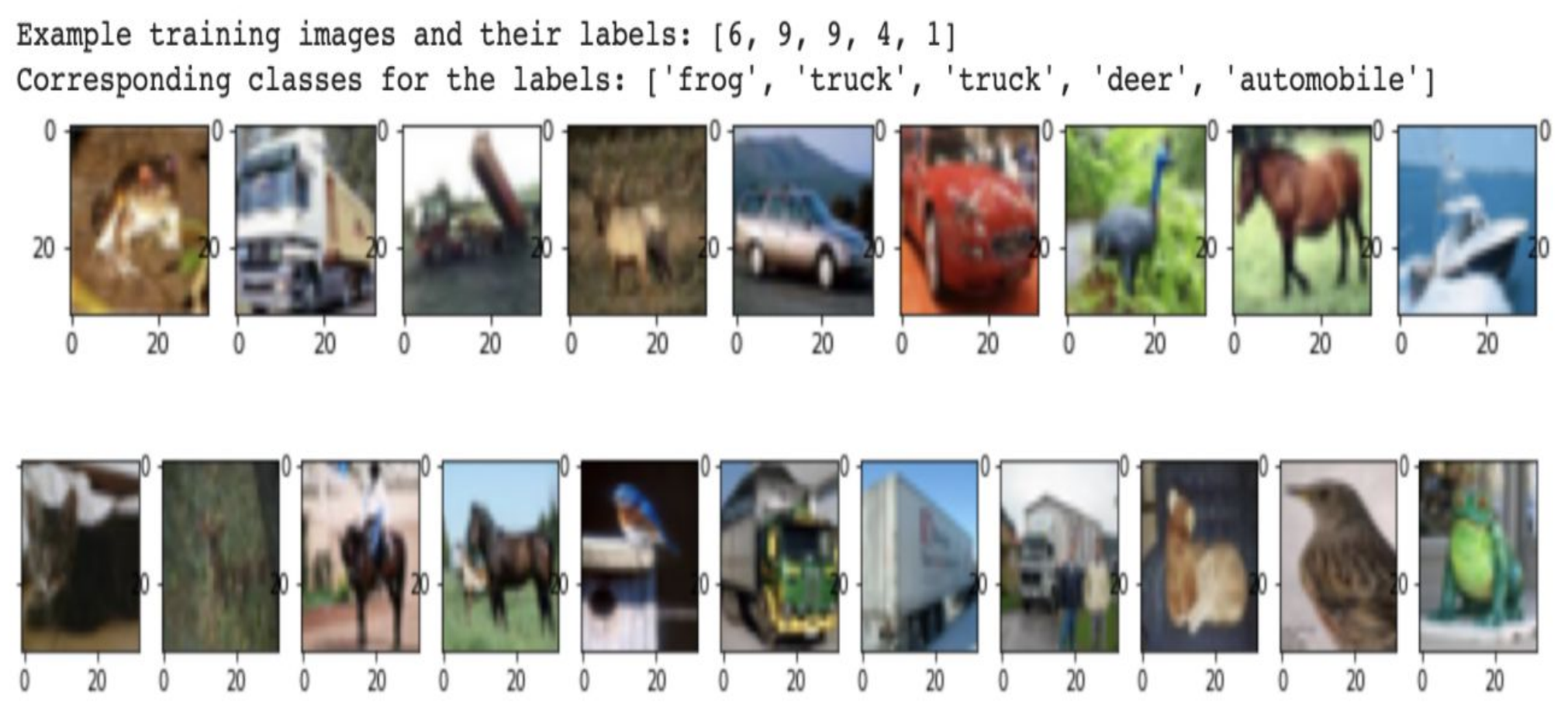
Algorithm Flow of NetAdapt

Lookup Tables for Fast Resource Estimation

Algorithm flow of NetAdapt

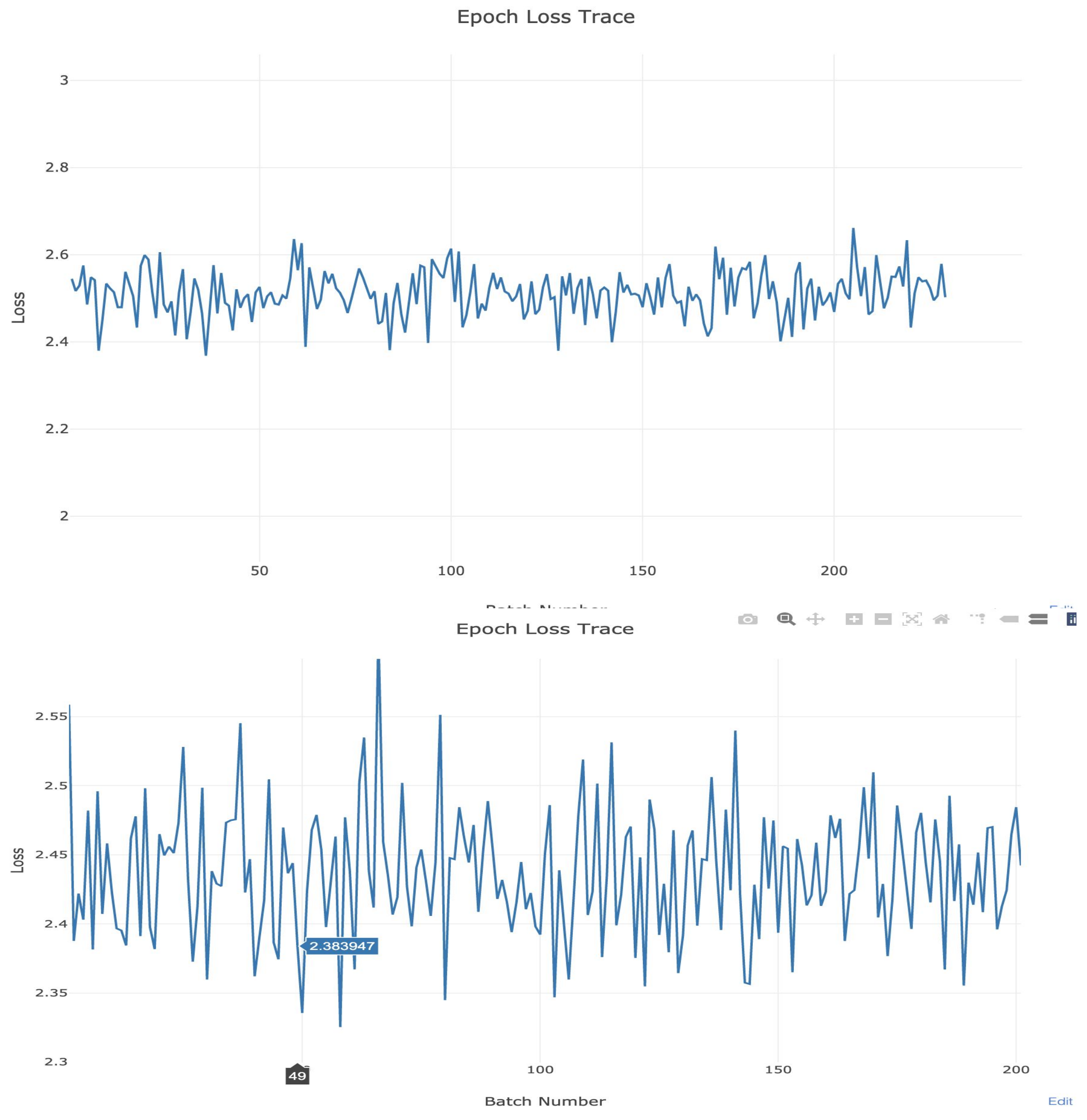
Analysis and Results

- CIFAR-10 and PKU-Autonomous Driving dataset were pre-processed which were further fed to each of the designed CNN models.
- Given any activation function, when a back-propagation algorithm is implemented in order to optimize the network, the larger values of input could raise the problem of exploding gradients or vanishing gradients which is not favorable for developing a best Neural Network model.

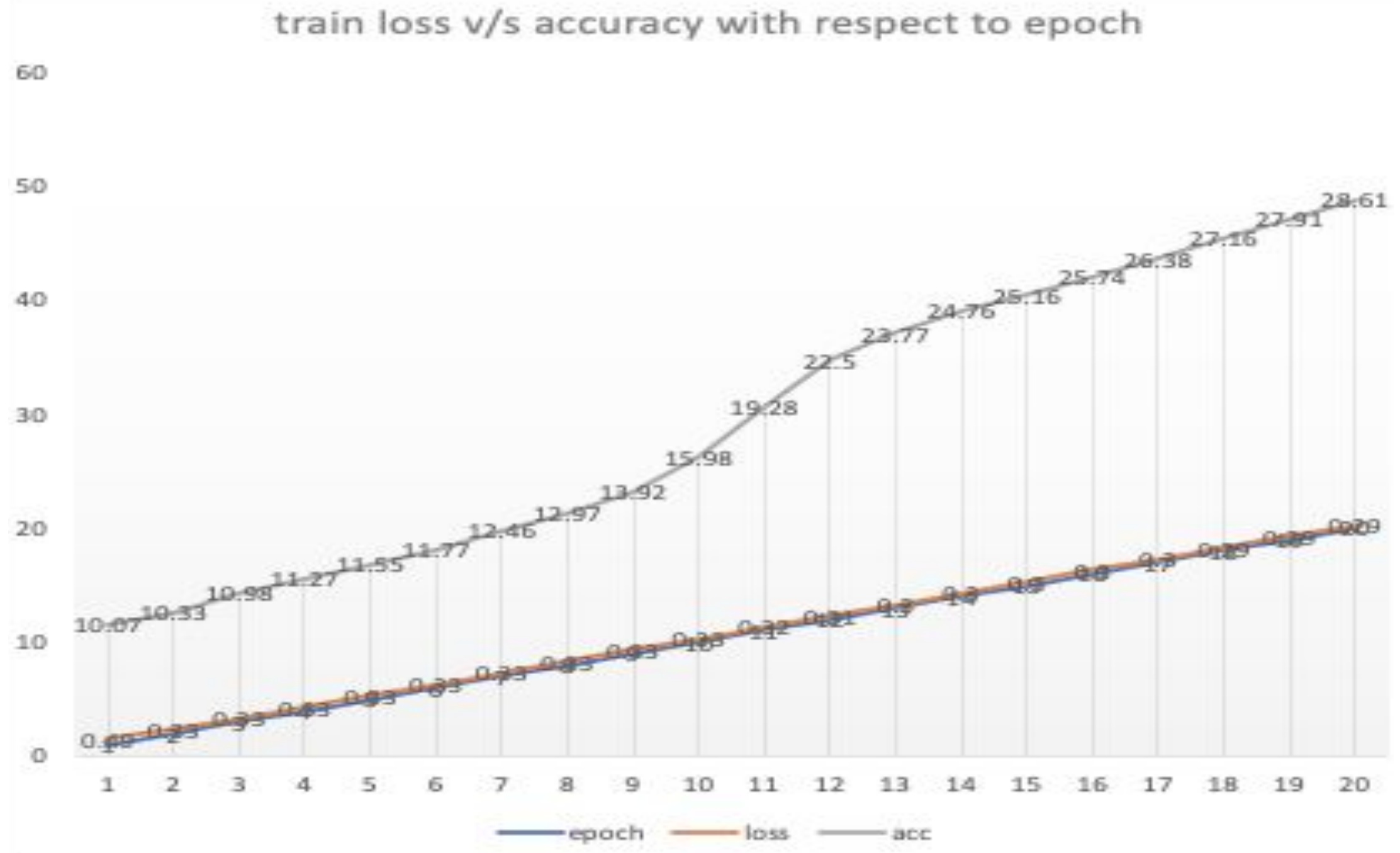


```
Example training images and their labels: [6, 9, 9, 4, 1]
Corresponding classes for the labels: ['frog', 'truck', 'truck', 'deer', 'automobile']
```

CIFAR10 Image Dataset and its Comparison Matrix

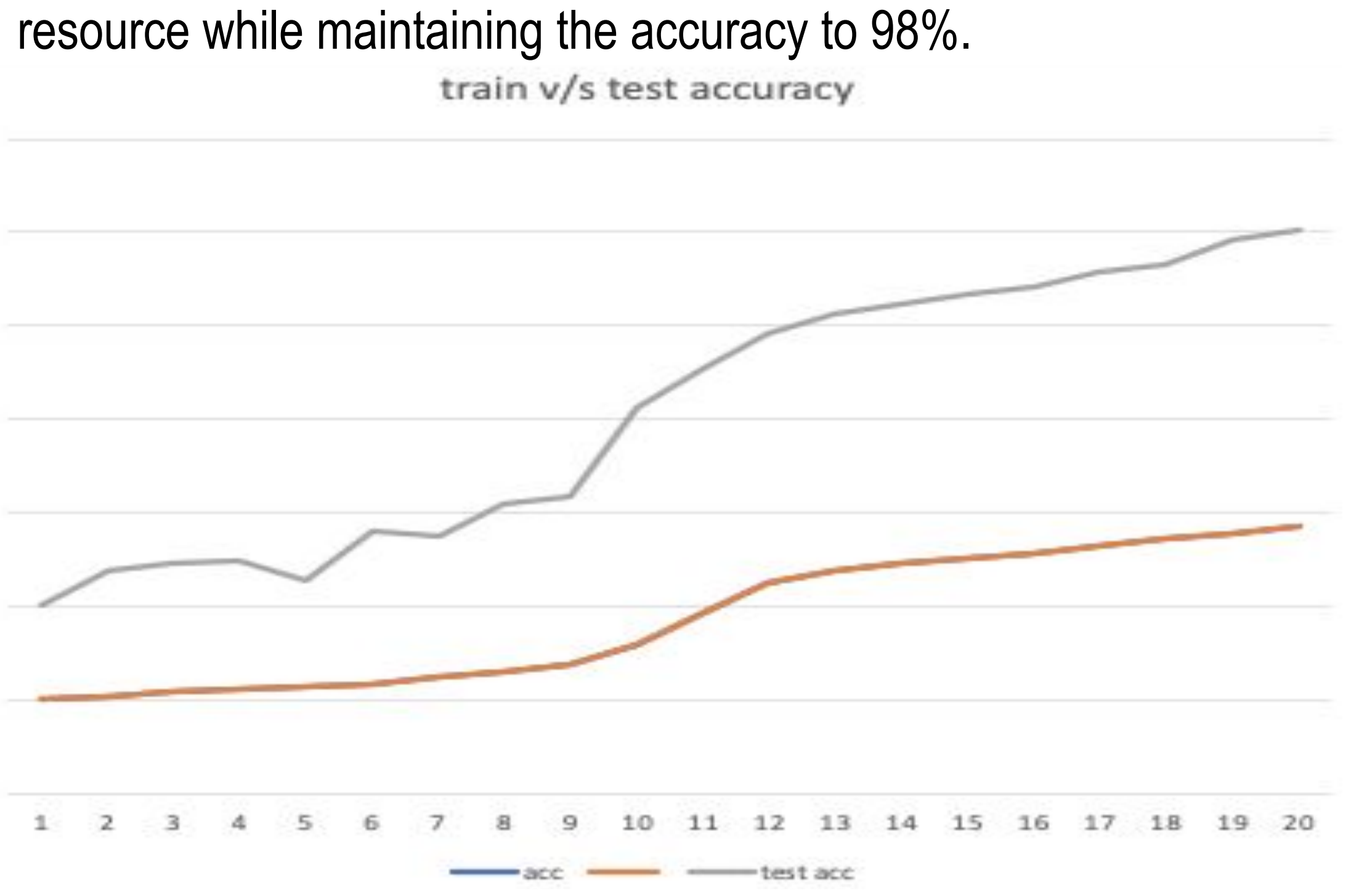


Quantized ResNet Training and Test Results



NetAdapt applied on AlexNet and its loss vs accuracy plot

- 13 iterations were performed and the last model requires less resource while maintaining the accuracy to 98%.



Train vs Test Accuracy for NetAdaot+AlexNet for CIFAR-10 dataset.

Summary/Conclusions

The model were successfully implemented and verified using AlexNet and ResNet while learning their behaviors for different datasets with more accuracy along with less energy/resource consumption and computational cost.

The Quantization of ResNet was implemented by reducing the weights from full-precision to 1-bit representation and the feature maps were represented as 8-bit values. NetAdaptat was applied on a pre-trained AlexNet model analyzing the result.

Key References

[1] Kunyuan Du, Ya Zhang, and Haibing Guan, "From Quantized DNNs to Quantizable DNNs", 11 April 2020.

[2] Tien-Ju Yang1, Andrew Howard , Bo Chen , Xiao Zhang , Alec Go , Mark Sandler, Vivienne Sze1 , and Hartwig Adam proposed "NetAdapt: Platform-Aware Neural Network Adaptation for Mobile Applications", EVCC 2018 paper.

[3] Y.-H. Chen T. Krishna J. Emer V. Sze "Eyeriss: An energy-efficient reconfigurable accelerator for deep convolutional neural networks" IEEE J. Solid-State Circuits vol. 51 pp. 127-138 Jan. 2017.

[4] Mingyu Gao, Xuan Yang, Jing Pu, Mark Horowitz, and Christos Kozyrakis proposed "TANGRAM: Optimized Coarse-Grained Dataflow for Scalable NN Accelerators", 2019 Architectural Support for Programming Languages and Operating Systems (ASPLOS '19), April 13–17, 2019, Providence, RI, USA. ACM, New York, NY, USA.

Acknowledgements

The authors are deeply indebted to Dr. Chang Choo for his invaluable their patience and professional advice on this project. We also appreciate the Electrical Engineering department to grant the opportunity for presenting this project.

www.PosterPresentations.com