ARL-TN-1006 • FEB 2020

DEVCOM ARMY RESEARCH LABORATORY

On Benchmarking Multiple GPU Computing Resources for Faster Training of Deep Neural Networks

by Arnold Tunick

NOTICES

Disclaimers

The findings in this report are not to be construed as an official Department of the Army position unless so designated by other authorized documents.

Citation of manufacturer's or trade names does not constitute an official endorsement or approval of the use thereof.

Destroy this report when it is no longer needed. Do not return it to the originator.

ARL-TN-1006 ● FEB 2020

On Benchmarking Multiple GPU Computing Resources for Faster Training of Deep Neural Networks

Arnold Tunick
Computational and Information Sciences Directorate, CCDC Army Research Laboratory

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED (From - To)
February 2020	Technical Note	October 2019 – January 2020

4. TITLE AND SUBTITLE
On Benchmarking Multiple GPU Computing Resources for Faster Training of Deep Neural Networks

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Arnold Tunick

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
CCDC Army Research Laboratory
ATTN: FCDD-RLC-IB
Adelphi, MD 20783

8. PERFORMING ORGANIZATION REPORT NUMBER
ARL-TN-1006

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
ORCID ID(s): A Tunick, 0000-0002-9228-1094

14. ABSTRACT
In this technical note, we explore a method to benchmark the performance of a Lambda Quad Deep Learning Workstation on training seven leading deep neural network (DNN) models using multiple GTX-1080 Ti graphics processing units (GPUs). We compute the average number of images processed per second for each DNN and quantify the consistent improvements in speed/performance when additional GPUs are assigned to the task. Our results show that the multi-GPU performance statistics calculated at the US Army Combat Capabilities Development Command Army Research Laboratory (CCDC ARL) are in close agreement to those calculated at Lambda Labs. The benchmarking of an accessible distributed computing resource provides an important step toward mitigating the excessive computational costs often associated with modern machine-learning applications, to include the training of DNNs on large data sets.

15. SUBJECT TERMS
distributed computing; computational benchmark; machine-learning; TensorFlow; synthetic data

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Arnold Tunick
Unclassified	Unclassified	Unclassified	UU	16	19b. TELEPHONE NUMBER (Include area code) (301) 394-1233

Contents

List of Figures

List of Tables

Acknowledgments

The author acknowledges the support of the US Army Combat Capabilities Development Command Army Research Laboratory. The author thanks Brian Jalaian (CCDC ARL) for providing insightful comments and technical feedback to improve the manuscript.

1. Introduction

Exploiting multiple graphics processing units (GPUs) for distributed computing in machine learning has enabled researchers to train deep neural network (DNN) models on large data sets faster and in a more computationally efficient manner.[1–4] In this technical note, we explore a method[5] to benchmark the performance of a Lambda Quad Deep Learning Workstation (Lambda Labs, San Francisco, California) on training seven leading deep neural network (DNN) models using up to four GTX-1080 Ti GPUs.

We compute the average number of images processed per second during 100 iterations of training and quantify the improvements in speed/performance when additional GPUs are assigned to the task (i.e., by calculating the speed-up in performance for two, three, and four GPUs). We notice an approximate linear *n-fold* image processing speed increase in neural network model training, where *n* is the number of GPUs assigned. Our results show that the multi-GPU performance statistics calculated at the US Army Combat Capabilities Development Command Army Research Laboratory (CCDC/ARL) are in close agreement to those calculated at Lambda Labs.

2. Benchmark Results

The Python benchmark library[6] used for this analysis was developed at Lambda Labs and implemented at ARL using TensorFlow[7], version 1.14. For each GPU and DNN combination, we ran a single training experiment and measured the average number of images processed per second over 100 iterations (i.e., processing 64–256 image frames 100 times), where the batch size is scaled depending on the number of GPUs assigned, and with a learning rate of 0.1. Following the methods prescribed by Lambda Labs[8,9], these benchmark experiments used synthetic data, as opposed to real data. Synthetic data[9] are images of random pixel colors created directly on the GPU once (at the beginning of the benchmark), so there is no reading data from disk or copying data from CPU memory to GPU memory, thus minimizing many non-GPU related bottlenecks. In addition, the multi-GPU training was performed using model-level distributed computing. Table 1 summarizes the benchmark results computed at ARL, showing data for both 16- and 32-bit floating-point precision (i.e., FP16 and FP32). These data show that the number of images processed per second consistently increases with the number of GPUs assigned to the task. Figure 1 presents a bar chart to illustrate the speed-up in performance statistics for FP32 for each DNN tested. As an example, it was found that the increase in the number of image frames processed per second for the ResNet-50 DNN was 1.61 and 3.73 times greater for two and four GPUs, respectively.

Table 1 Benchmark results for seven leading DDNs using the Lambda Quad Workstation and up to GTX-1080 Ti GPUs. Each column shows the measured number of images processed per second for each DNN tested.

Model	1 × GPU		2 × GPU		3 × GPU		4 × GPU	
	FP16	FP32	FP16	FP32	FP16	FP32	FP16	FP32
ResNet-50	272.32	216.25	454.72	349.08	677.40	537.13	1031.28	806.56
ResNet-152	103.02	86.15	171.88	147.75	265.56	216.08	396.62	324.72
Inception v3	159.53	139.42	276.20	216.82	377.12	348.79	628.82	533.83
Inception v4	66.43	60.27	106.10	97.59	147.33	136.05	243.37	216.61
VGG16	153.47	137.68	260.76	213.99	365.15	279.09	533.83	422.36
AlexNet	2969.45	2841.14	4514.61	3731.97	6782.30	5451.51	10625.52	8821.52
SSD300	128.64	112.22	210.78	185.90	338.84	288.19	501.59	422.61

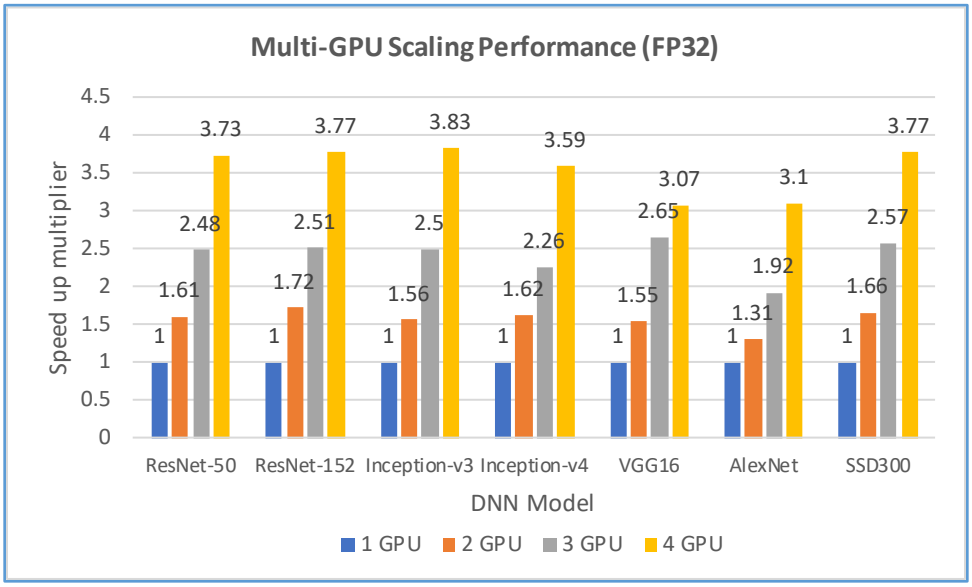

Fig. 1 Performance statistics calculated at ARL showing an approximate *n-fold* increase in images processed per second training speed

For comparison, the bar chart shown in Fig. 2 presents the performance statistics for six different high-end GPUs calculated in March 2019 at Lambda Labs, to include the GTX-1080 Ti GPU (shown on the far right side of the graph). An equivalent experimental method as described previously was applied by Lambda Labs with the exception that they measured the speed-up/performance for 10 training experimental trials with 1, 2, 4, and 8 GPUs on each neural network model and then averaged the results. Table 2 consolidates the GTX-1080 Ti data from Figs. 1 and 2, demonstrating that the performance statistics calculated at ARL are in close agreement to the data generated at Lambda Labs.

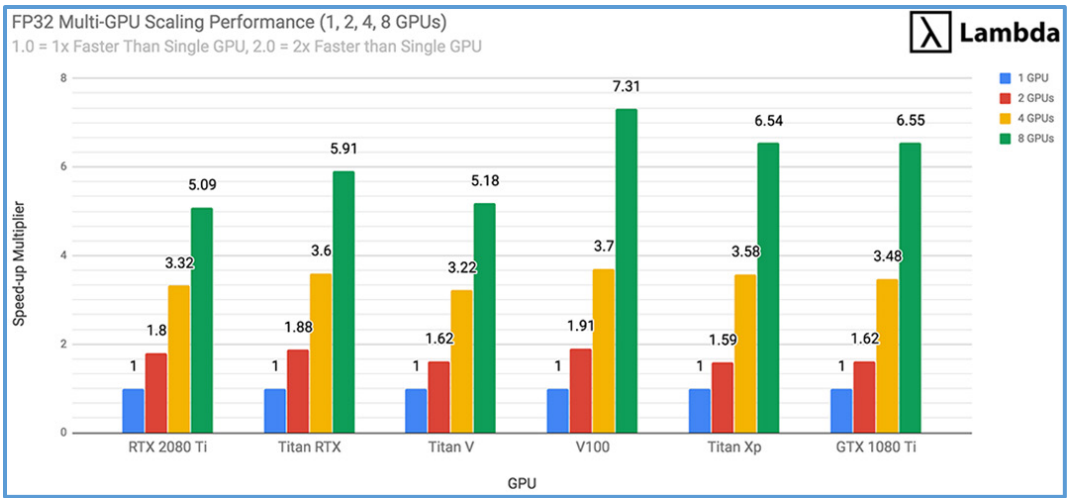

Fig. 2 Performance statistics calculated at Lambda Labs for six high-end GPUs to include the GTX-1080 Ti GPU shown on the far right side of the graph

Table 2 Comparison of performance statistics calculated at ARL versus those calculated at Lambda Labs

Case study	2 × GPU	4 × GPU
At ARL[a]	1.58	3.55
At Lambda Labs [b]	1.62	3.48

[a] Computed as the average value from a single experimental trial across all seven DNN models.
[b] Computed as the average value from 10 experimental trials across all seven DNN models.

In addition, Table 3 shows that the number of images processed per second calculated at ARL over a single experimental trial using 1 GPU are in close agreement to the data generated at Lambda Labs calculated as the average over 10 experimental trials.

Table 3 Comparison of benchmark results using 1 GPU. Each column shows the calculated number of images processed per second.

Model	FP16 bit		FP32 bit	
	ARL	Lambda Labs	ARL	Lambda Labs
ResNet-50	272.32	263	216.25	209
ResNet-152	103.02	96	86.15	81
Inception v3	159.53	156	139.42	136
Inception v4	66.43	62	60.27	58
VGG16	153.47	149	137.68	134
AlexNet	2969.45	2891	2841.14	2762
SSD300	128.64	123	112.22	108

3. Related Work

Advances in the development of deep learning neural networks have enabled state-of-the-art results in machine learning for an ever-growing array of military and domestic applications. Unfortunately, these machine-learning models have been shown to be vulnerable to specially crafted adversarial perturbations and manipulation, which can result in machine learning systems being easily fooled or subverted. For real-world battlefield information systems, in particular, adversarial attacks can cause a target of interest to be misclassified, making something that is a threat appear as benign or vice versa. Such a scenario creates a significant challenge in deploying deep learning models in security-critical domains where adversarial activity is most prevalent, such as Internet of Battle Things (IoBT), cyber-networks, and surveillance. To better understand adversarial attacks in the context of battlefield information systems, Tunick and Jalaian[10,11] presented a review of the current literature in adversarial machine learning (AML). They highlighted many open challenges in the area of AML to be addressed by the research community to help to ensure increased resiliency of autonomous systems in contested environments. In addition, they reported that CCDC ARL and our IoBT special project collaborators have developed a relatively new class of detection algorithms to differentiate between clean and attacked images providing a potentially viable defense to adversarial manipulation of battlefield data.[12,13]

Hence, our increasing efforts to exploit multiple GPUs for distributed computing in machine learning could enable CCDC ARL and our IoBT colleagues to train DNNs on large datasets faster and generate adversarial examples in a more computationally efficient manner for the development and testing of novel adversarial attack detection methods.

With regard to other established distributed computing benchmarks for multi-GPU computing, the DAWNBench Deep Learning Benchmark[2,3] currently provides a list of the best/fastest DNN training times for the ImageNet data set.[14] Within DAWNBench, DNNs such as ResNet-18 and ResNet-50 were trained to a specified top-5 accuracy of 93% rather than measuring the individual images per second processing speeds. The DAWNBench threshold of 93% appears to be a competitive threshold achievable through specially crafted optimizations of distributed multi-GPU or multi-node computing, as well as adjustments of key hyperparameters, such as batch size and learning rate. For the evaluation of neural network models, the top-5 accuracy rate is defined as the fraction of test images for which the correct class label is among the five most likely class labels determined by the model.[1]

4. Summary and Conclusion

We explored a method to benchmark the performance of a Lambda Quad Deep Learning Workstation on training seven leading DNNs using the TensorFlow library framework. We computed the average number of images processed per second for each DNN and showed that these values increased linearly, approximately *n-fold*, where *n* is the number of GPUs assigned to the task. Moreover, the statistics for the number images processed per second calculated at ARL for the GTX-1080 Ti GPU were shown to be in close agreement to the values generated at Lambda Labs for an equivalent experimental implementation. Our benchmarking an accessible distributed computing resource at CCDC ARL provides an important step toward mitigating the excessive computational costs often associated with modern machine-learning applications, to include the training of DNNs on large data sets.

5. References

1. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. Proc. 25th International Conference on Advances in Neural Information Processing Systems; 2012.

2. Coleman C, Narayanan D, Kang D, Zhao T, Zhang J, Nardi L, Bailis P, Olukotun K, Re C, Zaharia M. DAWNBench: an end-to-end deep learning benchmark and competition. Proc. 31st Conference on Neural Information Processing Systems; 2017.

3. DAWNBench: an end-to-end deep learning benchmark and competition. Stanford (CA): Stanford University; 2020 [accessed 2020 Jan 7]. https://dawn.cs.stanford.edu/benchmark.

4. Patterson D. MLPerf: SPEC for ML. Berkeley (CA): University of California Berkeley; 2018 May 2 [accessed 2019 Dec]. https://rise.cs.berkeley.edu/blog/mlperf-spec-for-ml/.

5. RTX 2080 Ti deep learning benchmarks with TensorFlow; 2019 Mar 4 [accessed 2020 Jan]. https://lambdalabs.com/blog/2080-ti-deep-learning-benchmarks/.

6. Lambda TensorFlow benchmark. GitHub; 2018 Nov 28 [accessed 2020 Jan]. https://github.com/lambdal/lambda-tensorflow-benchmark.

7. Toleubay Y, James AP. Getting started with TensorFlow deep learning. In: James A, editor. Deep learning classifiers with memristive networks. New York (NY): Springer, Cham; 2020. (Modeling and optimization in science and technologies, vol. 3).

8. Agrawal M. Lambda Labs, San Francisco, CA. Personal communication, 2020 Jan.

9. Sarkar T. Synthetic data generation — a must-have skill for new data scientists. 2018 Dec 18. [accessed 2020 Jan]. https://towardsdatascience.com/synthetic-data-generation-a-must-have-skill-for-new-data-scientists-915896c0c1ae.

10. Tunick A, Jalaian B. A review of adversarial attacks in the context of battlefield information systems. IEEE MILCOM; 2019; Norfolk, VA.

11. Tunick A, Jalaian B. On adversarial machine learning for battlefield information systems. Adelphi (MD): CCDC Army Research Laboratory (US); 2019 Aug. Report No.: ARL-TR-8757.

12. Jha S, Jang U, Jha S, Jalaian B. Detecting adversarial examples using data manifolds. IEEE MILCOM; 2018; Los Angeles, CA.

13. Jha S, Raj S, Fernandes S, Jha SK, Jha S, Jalaian B, Verma G, Swami A. Attribution-based confidence metric for deep neural networks. Proc. 33rd Conference on Neural Information Processing Systems. 2019.

14. Russakovsky O, Deng J, Su H, Krause J, Satheesh S, Ma S, Huang Z, Karpathy A, Khosla A, Bernstein M, et al. ImageNet large scale visual recognition challenge. Int J Comput Vis. 2015;115(3):211–252.

List of Symbols, Abbreviations, and Acronyms

AML	Adversarial Machine Learning
ARL	Army Research Laboratory
CCDC	US Army Combat Capabilities Development Command
DNN	Deep Neural Network
GPU	graphics processing unit
IoBT	Internet of Battlefield Things

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
1 (PDF)	CCDC ARL FCDD RLD CL TECH LIB
7 (PDF)	CCDC ARL FCDD RLC I S RUSSELL R HARDY FCDD RLC IB S LAROCCA L HERNANDEZ J FREEMAN A TUNICK B JALAIAN