Effects of Overclocking on the Performance of Graphics Processing Units in AI-Based Image Classification

Youjun (Fred) Han
New Hyde Park Memorial High School
New Hyde Park, New York, United States of America
yhan84273@tutanota.com

***Abstract*—Machine learning (ML), more specifically deep learning has been on a consistent rise over the last two decades. An aspect of ML that has recently seen significant performance improvements lies in image classification. Although highly helpful for the world as it is capable of providing near real-time responses with high accuracy, image classification back-end software is often highly power intensive due to its reliance on graphics processors to process the images and do the true classification in a series of convolutions. With this in mind, we attempted to test the performance of graphics processors when overclocked or undervolted to test the impact these modifications will have on the accuracy and inference time using the ResNet50 image classification framework. It is hypothesized that overclocking will improve the inference time of image classification with similar accuracy to the baseline at the cost of higher instantaneous power consumption whereas the undervolted processors will cause increased inference time, with similar accuracy, but with significantly lower instantaneous power consumption. This project was conducted on Windows 11 with WSL 2 using the TensorFlow machine learning library with Nvidia graphics processors utilizing CUDA to improve the processing speeds of this process using graphics hardware. It was concluded that overall, the performance deltas between overclocked, undervolted, and stock processors were within a margin of error. All trials on the RTX 3080 resulted in an average power draw of approximately 85 W, a total time average of approximately 350 seconds per training run, and an accuracy of approximately 85% regardless of the training conditions the tested Nvidia RTX 3080 graphics processor was under. When overclocked, the RTX 3080 had a best-case training speed increase of 8.3% over undervolted execution. The RTX 3060 trails resulted in average power draws of approximately 50 W with total time averages of approximately 330 seconds per training run with similar accuracy to the RTX 3080. When overclocked, the RTX 3060 had a best-case single epoch training speed increase of 50% over the nominal performance of 46 seconds compared to 21 seconds.**

***Index Terms*—Computer graphics, Image classification, Residual neural networks**

I. Introduction

A. Motivation

With the increasing computational power of processors in the modern day, there is a steady increase in power consumption estimates. With transistor count and density constantly on the rise, the trend of increased power consumption from modern processors will continue upward as a result of either variable. Transistor density has recently improved due to Intel RibbonFET and transistor count will almost certainly continue to increase, just with a larger die size. Thus, there exists reasons to improve computer computation efficiency It is the purpose of this project to test the viability of undervolting and overclocking on modern Graphics Processing Units (GPUs) for use in ML applications.

When considering the ever-growing power demands of processors and the growing calls for sustainable energy production and consumption, the necessity to improve processor processing times and power consumption has increased significantly. With there being existing research conducted by Microsoft, an extremely large cloud services provider, in overclocking servers in cloud environments that show improvements in latency and container density [7], [8], the viability of this project is rather high.

II. Background

Image Classification in modern Artificial Intelligence (AI) work has significantly improved in the past decade. Python libraries such as TensorFlow and PyTorch have made all types of machine learning more accessible to the general public than ever with their simple syntax and short setups. With the accessibility of AI to the average consumer growing, there is an increasing demand for computing power. Although this technology has significantly improved due to the increase in research done, it is rather power-intensive. Whether a user runs ML work on personal computers or in the cloud using a service such as Google Colab, efficiency is often rather low with Power Usage Effectiveness (PUE) ranging from 1.1 in cloud-based computing environments to 2.5 in local data centers hosted by individual companies [1]. Data centers such as those run by Google often perform significant ML tasks run by clients. These data centers are estimated to need 321 Terawatt hours (TWh) of power per year to operate in 2030 [2]. As Moore's Law previously stated, there would be an approximate doubling of transistors every year in computer processors [6]. Although this claim has lost validity due to the theoretical limit of transistor density drawing nearer [5], the concern of transistor density translating to increased power consumption

largely stands true. From 231 Watts (W) average when gaming on the Nvidia GTX 1080 Ti in 2017 to 346 W average when gaming on the RTX 4090 in 2022, the power requirements of modern GPUs have significantly increased in the past decade [3], [4]. With this increase in energy needs, the necessity for improved sustainable computing grows accordingly.

Overclocking is the act of increasing a processor's clock speed. This is often done through software that processors are bundled with such as Intel Extreme Tuning Utility (XTU), AMD Ryzen Master, or MSI Afterburner. In the case of embedded devices, they are hard programmed using Software Development Kits (SDKs) to increase clock speed artificially.

Fig. 1. MSI Afterburner overclocking utility

Similarly, undervolting is the act of decreasing a processor's maximum voltages to decrease power consumption. This could be done using the same tools as overclocking in most scenarios. Through undervolting, performance is often stunted, however, depending on the architecture undervolting could be potentially beneficial. Through power savings, the thermal limits of the processor will be more difficult to reach in addition to running cooler at idle power [9].

In the case of our testing, we allowed the processors to continue boosting at higher voltages, past our undervolted voltages using a curve. The curve was generated through MSI Afterburner and prior to a significant increase in clock, the voltage will be significantly lower than what was manufacturer set. Once the clock of the GPU reaches the set setting, it will boost and increase voltages accordingly using MSI Afterburner.

III. Literature Review

A. Overclocking and Undervolting

Significant research has been done on the performance improvements and drawbacks of overclocking and undervolting. Several studies have investigated the impact of these techniques on various types of processors. In a 2018 study conducted by Thomas and Shanmugasundaram, different overclocking techniques utilize different processors and boards [13]. It was shown that there were some performance uplifts and positives to processor overclocking, however, there were also downsides such as shortened lifetimes, etc.

Additionally, in a study funded by Microsoft conducted by Jalili et al., overclocking was considered as an option for use in data centers where raw performance is extremely significant [7]. It was found by the researchers that overclocking did indeed improve performance in the data centers, however, there was also a significant disparity in PUE of a data center based on what cooling solution it most utilized. For instance, data centers utilizing liquid cooling had a significantly better PUE than those using air cooling. The researchers concluded that while overclocking does significantly improve performance, the implementation and effectiveness of such a technique are highly dependent on the cooling solutions in place. This highlights the importance of adequate cooling systems and power management in overclocked environments.

B. Neural Networks and Residual Neural Networks

There have been some studies conducted in the realm of neural networks and their performance. In a 2022 article by Asam et al., different neural network frameworks were tested to identify malware based on image representations of software code to detect malware [10]. In this study, the tested neural networks were tested to find the most performant neural network framework for doing this task. It was shown that the most performant neural network framework was the new novel framework built by the researchers, however, one of the highest performing existing TensorFlow frameworks was ResNet which was used as the framework for our project.

The 2015 paper pertaining to the making of ResNet conducted by He et al., outlined the issues of training extremely deep neural networks and introduced the concept of residual learning to address these issues [14]. The authors proposed a simple yet effective architecture design, which won the 1st place on the ILSVRC 2015 classification task. They showed that their designed network, ResNet, can be easily trained with a large number of layers, up to 152 layers, which is significantly deeper than previous networks.

IV. Methodology

This project's main purpose is to be able to compare the performance of non-stock consumer GPUs against the performance of factory-setting GPUs in training ResNet50 models. The performance will be measured using deltas between stock, overclocked at 1900 MHz, undervolted at 1900 MHz, and undervolted at 2100 MHz. The research methodology of this paper is split into six stages.

First, we conducted the initial literature review. This was primarily used to visualize the viability of our project in addition to seeing testing on similar models.

Following the literature review, we assembled the computer that was to be used. The assembly of the computer followed ISO 29136 standards. For monitoring purposes, MSI Afterburner, Ryzen Master, and HWINFO64 have been installed on the Windows 11 installation.

Once assembly was complete, we began programming the machine learning algorithm to be tested. The programming was fully done in the Python programming language with Tensorflow being the Machine Learning algorithm of choice. This was accompanied by Nvidia CUDA Version 12.2.128 and Nvidia Driver Version 537.13. The program is primarily based on the OVHcloud AI Training Examples ResNet50 file with minor modifications [11]. The initial dataset used is the Flower Classification dataset from Mendeley Data [12].

Computer Specifications

CPU	AMD Ryzen 9 3950X	
Motherboard	Gigabyte B550i Aorus Pro AX	
RAM	GSkill F4-4000C18D-32GVK	
Power Supply	XPG Core Reactor 850W	
GPU	MSI Nvidia RTX 3080 Ventus 3X	EVGA Nvidia RTX 3060 XC Gaming
Storage	Intel 660P 512GB	WL4000GSA6472E
Operating System	Windows 11 Pro 22H2	
Windows Subsystem for Linux 2	Debian Version 11	

Once programming was complete, data collection was relatively straightforward. We conducted 10 trials of each setting with HWINFO64 logging power, clock, temperature, and load percentage. The amount taken for a training session, average amount of time per epoch, and accuracy were recorded using the built-in TensorFlow logging software.

Fig. 2. HWINFO64 logging utility.

Tested Settings (MSI Nvidia RTX 3080 Ventus 3X)

Modification	Clock Speed	Voltage
Stock	1710 MHz	1.069 V
Overclocked	1810 MHz	1.063 V
Undervolted	1210 MHz	0.950 V
Undervolted	1890 MHz	0.950 V
Undervolted	2010 MHz	0.950 V

Finally, we used the collected data from HWINFO64 in CSV form and analyzed them in Microsoft Excel.

V. Results and Discussion

In order to significantly assess the performance deltas between the different settings, we conducted 10 instances of each setting. The averages of the 10 instances including accuracy, average time, total time of training in addition to power, clock, temperature, and average load percentage of the GPUs are listed below.

Tested Settings (EVGA Nvidia RTX 3060 XC Gaming)

Modification	Clock Speed	Voltage
Stock	1950 MHz	1.081 V
Overclocked	2050 MHz	1.081 V
Undervolted	1450 MHz	1.081 V
Undervolted	2050 MHz	1.081 V
Undervolted	2150 MHz	1.081 V

Results (MSI Nvidia RTX 3080 Ventus 3X)

Type	Power (W)	Clock (MHz)	Temp. (°C)	Load (%)	Total Time (s)	Average Time (s)	Accuracy (%)
Stock, 1710 MHz, 1.069 V	87.097	892.6	61.022	32.884	348	23.206	0.854
Overclocked, 1810 MHz, 1.063 V	85.741	795.7	61.573	33.169	350	23.327	0.856
Undervolted, 1210 MHz, 0.950 V	85.095	802.6	60.355	33.623	339	23.345	0.859
Undervolted, 1890 MHz, 0.950 V	87.632	939.2	61.304	32.894	350	23.352	0.850
Undervolted, 2010 MHz, 0.950 V	88.018	931.9	61.770	33.175	349	23.300	0.852

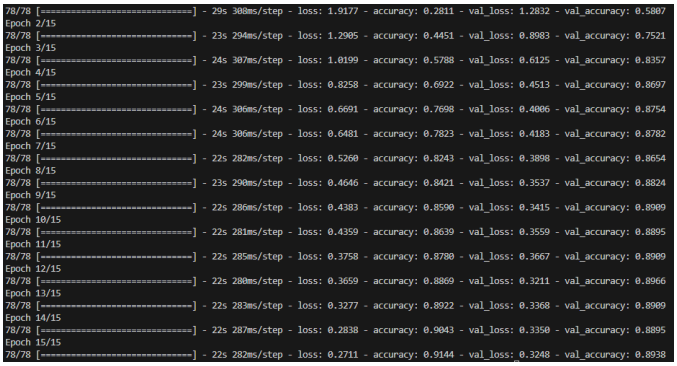

Fig. 3. 15 epoch run of the RTX 3080 at manufacturer settings.

The results show that performance was rather constant on average throughout all of the trials, with all variables being within a margin of error of one another. Power, temperature, load percentage, average time per epoch, and accuracy were rounded to the nearest thousandth. Clock was rounded to the nearest tenth, and total time for 15 epochs was rounded to the nearest digit. We considered a tolerance for insignificant results at 5% and these results were generally within this range, thus considered to be within a margin of error. However, when overclocked, there is a notable difference on the RTX 3080 in the best performance on the overclocked configuration as compared to the worst performance on the undervolted and factory configuration at approximately the difference of one epoch of 33 seconds.

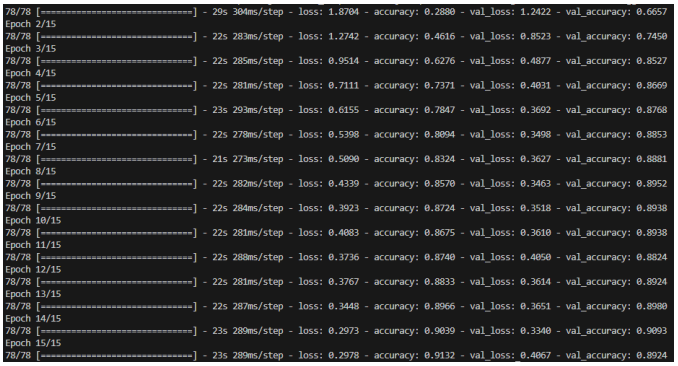

Fig. 4. 15 epoch run of the RTX 3080 at 100 MHz over stock (1810 MHz) when overclocked.

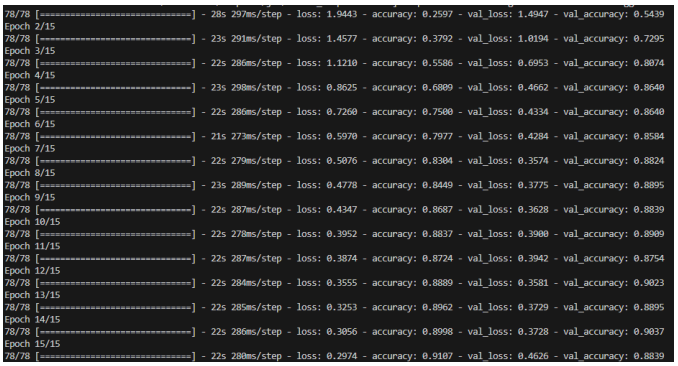

Fig. 5. 15 epoch run of the RTX 3080 at 200 MHz over stock (1890 MHz) when undervolted.

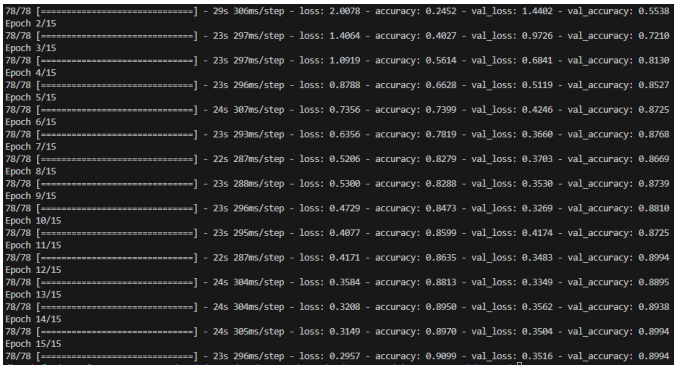

Fig. 6. 15 epoch run of the RTX 3080 at 300 MHz over stock (2010 MHz) when undervolted.

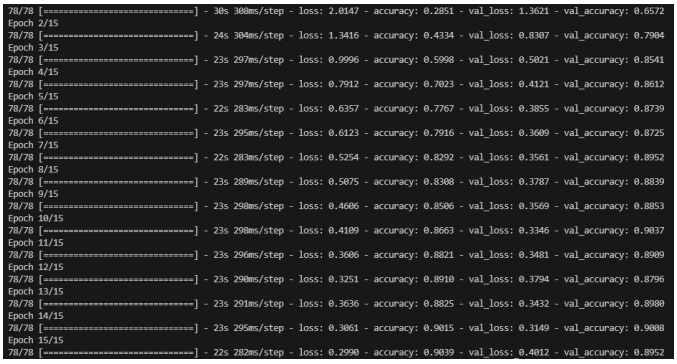

Fig. 7. 15 epoch run of the RTX 3080 at 500 MHz below stock (1210 MHz) when undervolted.

Results (EVGA Nvidia RTX 3060 XC Gaming)

Type	Power (W)	Clock (MHz)	Temp. (°C)	Load (%)	Total Time (s)	Average Time (s)	Accuracy (%)
Stock, 1950 MHz, 1.081 V	50.865	1519.2	53.645	34.344	334	22.247	0.853
Overclocked, 2050 MHz, 1.081 V	51.496	1573.4	53.536	34.049	334	22.287	0.855
Undervolted, 2050 MHz, 0.950 V	51.522	1584.7	53.494	35.520	330	22.022	0.851
Undervolted, 2150 MHz, 0.950 V	51.131	1493.7	54.832	34.148	342	22.81	0.857
Undervolted, 1450 MHz, 0.950 V	51.457	1082.8	53.987	32.606	334	22.30	0.847

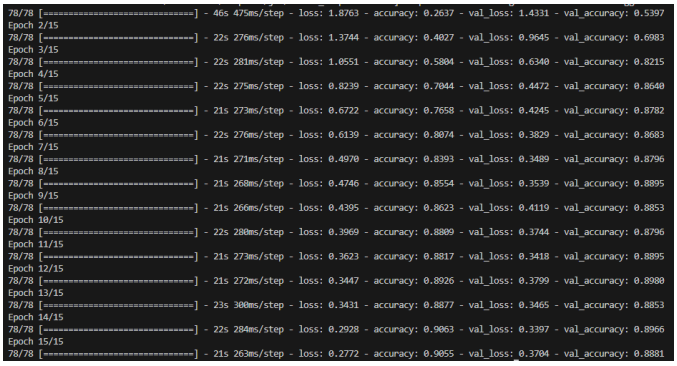

Fig. 8. 15 epoch run of the RTX 3060 at manufacturer settings.

The RTX 3060 generally had single run times faster than the RTX 3080 at around 22 seconds per epoch regardless of the setting it was used on. Undervolting provided a worst case scenario single epoch speed of 46 seconds whereas the best epoch was completed in 21 seconds.

Additionally, it was noted that the EVGA Nvidia RTX 3060 XC Gaming outperformed the MSI Nvidia RTX 3080 Ventus 3X in speed on multiple occasions. Despite the superior raw performance of the RTX 3080, the RTX 3060's increased memory capacity (12 GB instead of 10 GB) made it the stronger candidate for ML work in this scenario.

Notably, the load percentages of all runs were rather low at approximately between 30% and 50% GPU Core load. This was noted as a GPU optimization and scheduling expectation by Tensorflow. This could potentially explain why average performance was similar when the GPUs were trained on the same dataset even with different clock configurations.

Also, it was rather surprising and helpful to know that maximum core undervolts yielded similar results to nominal and overclocked performance in this case. Knowing this, we can discern that power can be limited without a significant detriment to performance. Power limits were kept constant in

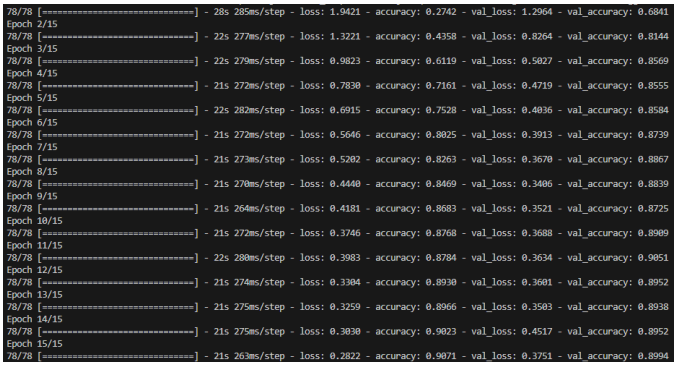

Fig. 9. 15 epoch run of the RTX 3060 at 100 MHz over stock (2050 MHz) when overclocked.

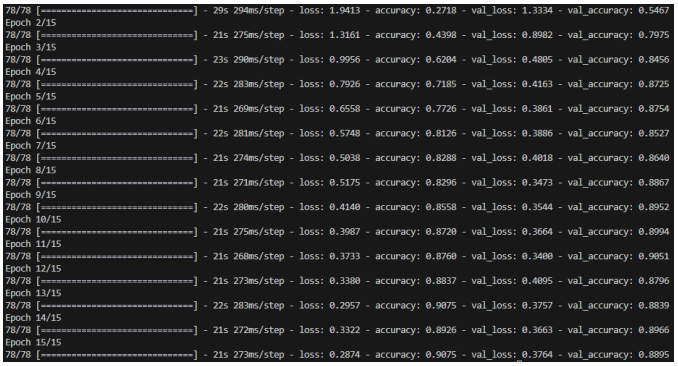

Fig. 10. 15 epoch run of the RTX 3060 at 100 MHz over stock (2050 MHz) when undervolted.

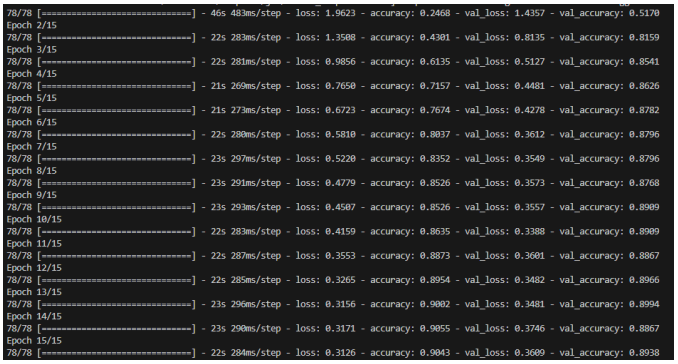

Fig. 11. 15 epoch run of the RTX 3060 at 200 MHz over stock (2150 MHz) when undervolted.

this experiment to prevent the chance of too many confounding variables from occurring, but may be tested in the future.

In general, the results of these tests were unexpected and contradict our hypothesis that ML training performance is bound by processor speeds. This could be due to a variety of reasons, however, most notably, undervolting the processors did not increase training time nor decrease accuracy in a significant manner, which stands to be beneficial to the end user. By significantly lowering the performance targets of the processor at idle and at low load, the processor's lifespan will be improved since it is not drawing as much voltage.

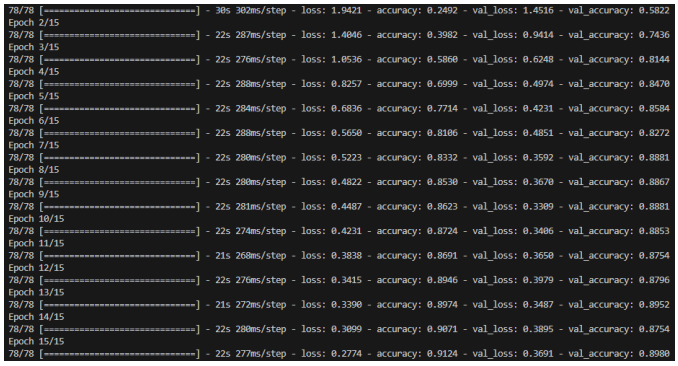

Fig. 12. 15 epoch run of the RTX 3060 at 500 MHz below stock (1450 MHz) when undervolted.

A. Future Work

There is considerable space for further evaluations and testing to ascertain the true effectiveness of overclocking with use in production environments. It would be incredibly important to improve the novelty of this project, especially in scope and hardware modification.

Specifically, more GPUs need to be tested under similar conditions, and other factors such as different deep learning architectures and different deep learning frameworks should be considered.

In particular, AMD GPUs that utilize ROCm technology such as the MI250 could be a strong candidate for further testing. This could help in understanding the performance characteristics of AMD GPUs in contrast to Nvidia GPUs in machine learning tasks.

It could also be worth revisiting AMD consumer GPUs once TensorFlow ROCm support for them is made official. TensorFlow currently does not support ROCm GPUs whereas PyTorch does. This disparity in support leads to a gap in knowledge regarding the performance of AMD GPUs in a TensorFlow environment.

Another aspect that could be explored is the performance of these GPUs in a multi-GPU setup. This could provide insights into the scalability of the system and how well these GPUs perform when combined. It would also be beneficial to see how the power consumption and thermal performance of the system can change within such a setup.

Furthermore, other aspects like the impact of different cooling solutions on performance and hardware longevity, and the effect of power limits on GPU performance could also be explored. Experiments could also be conducted to understand how different GPU architectures respond to overclocking and undervolting.

Additionally, the inference time and other ML settings should also be evaluated to provide a more complete picture. Energy efficiency and cost-effectiveness of the different configurations should also be taken into account to provide a more comprehensive analysis.

Testing the GPUs' performance in a distributed system setting is also a potentially valuable area of exploration, as

it may provide insights into how these configurations perform in real-world, large-scale applications.

B. Limitations

There are several limitations associated with this overall project that should be considered. Firstly, overclocking is often associated with hardware deterioration, especially in the case of heavily increased voltages. It is possible that although there are short-term gains in performance, the hardware may be significantly damaged in the process and decrease the longevity of the hardware, thus increasing operational costs.

A significant shortcoming of this project was that it was unable to test more GPUs. Although it was intended to test on ROCm using AMD GPUs, the lack of support for ROCm especially on newer architectures made it a difficult task to pursue.

Additionally, there is room for further testing for statistical significance. Due to time constraints, the results of this project cannot be considered statistically significant. Further studies pertaining to the performance of overclocked data center hardware must be done to ensure the significance of this project.

Furthermore, our testing only incorporated 1 graphics processor per model. This proved to be problematic since every individual processor has a different overclocking limit and will perform differently, accordingly. It is advised to use more processors of the same model in further testing for statistical significance.

Also, more testing could have been done to test other GPU related settings such as Power Limit, Core Voltage, Memory clock, etc. There is also a possible testing viability of shunt modding, etc.

VI. Conclusion

Through our experimental trials, we found that there were no differences between the performance of a GPU under different conditions, between stock, overclocked, and undervolted. This does contradict the common perception that overclocking improves performance, however, is not necessarily novel in that this has been found to be the case by other researchers. It is however, significant that underclocking does not change the performance of the training significantly.

In the context of machine learning tasks, it would seem that GPU performance is not entirely dependent on clock speed. Rather, it might be beneficial for future researchers to intentionally downclock their GPUs significantly for power efficiency in large data centers , without sacrificing much of the accuracy or training time.

In conclusion, our findings challenge the common perception that overclocking necessarily boosts GPU performance in machine learning tasks. Instead, we found that underclocking can potentially lead to comparable performance, while providing benefits in terms of power efficiency and hardware longevity. This could have significant implications for the operation of large-scale data centers, and for the future of machine learning research.

Acknowledgment

I'm extremely grateful to Pioneer Academics as it provided me with the resources to conduct this project and for connecting me with Dr. Seda Memik. In a similar vein, I would also like to express my most heartfelt thanks to Dr. Seda Memiik. She recommended the project to me upon our first individual meeting and helped me understand the importance of sustainable computing. In addition, Dr. Memik continued to support me wherever necessary, including sending research materials and helping with technical difficulties. Finally, I would like to thank Luke Smith of the University of Gloucestershire for his support in the technical aspects of this project.

References

[1] E. Masanet et al., 'The Energy Efficiency Potential of Cloud-Based Software: A U.S. Case Study', 6 2013.

[2] M. Koot and F. Wijnhoven, "Usage impact on data center electricity needs: A system dynamic forecasting model," Applied Energy, vol. 291, p. 116798, 2021. doi:10.1016/j.apenergy.2021.116798

[3] "NVIDIA GeForce GTX 1080 Ti Founders Edition 11 GB Review," Techpowerup, https://www.techpowerup.com/review/nvidia-geforce-gtx-1080-ti/.

[4] "Nvidia GeForce RTX 4090 Founders Edition Review - Impressive Performance" TechPowerUp, https://www.techpowerup.com/review/nvidia-geforce-rtx-4090-founders-edition/.

[5] D. Burg and J. H. Ausubel, 'Moore's Law revisited through Intel chip density', PLOS ONE, vol. 16, no. 8, pp. 1–18, 08 2021.

[6] R. R. Schaller, 'Moore's law: past, present and future', IEEE Spectrum, vol. 34, no. 6, pp. 52–59, 1997.

[7] M. Jalili et al., 'Cost-Efficient Overclocking in Immersion-Cooled Datacenters', in 2021 ACM/IEEE 48th Annual International Symposium on Computer Architecture (ISCA), 2021, pp. 623–636.

[8] P. A. Misra et al., 'Overclocking in Immersion-Cooled Datacenters', IEEE Micro, vol. 42, no. 04, pp. 10–17, Jul. 2022.

[9] "How undervolting can improve the performance of Nvidia Ampere series GPUs." Chaos, https://support.chaos.com/hc/en-us/articles/4419250847505-How-undervolting-can-improve-the-performance-of-Nvidia-Ampere-series-GPUs/

[10] M. Asam et al., 'IoT malware detection architecture using a novel channel boosted and squeezed CNN', Scientific Reports, vol. 12, no. 1, p. 15498, Sep. 2022.

[11] "Examples of use of OVHcloud AI Solutions" OVHcloud, https://github.com/ovh/ai-training-examples

[12] K. C. Tung, 'Flower Images jpg'. Mendeley, 2020.

[13] D. Thomas and M. Shanmugasundaram, 'A Survey on Different Overclocking Methods', in 2018 Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2018.

[14] K. He, X. Zhang, S. Ren, and J. Sun, 'Deep Residual Learning for Image Recognition'. arXiv, 2015.