Scaling MLPerf™ Inference vision benchmarks with Qualcomm Cloud AI 100 accelerators

Arjun Suresh
Krai Ltd
United Kingdom
arjunsuresh1987@gmail.com

Gavin Simpson
Krai Ltd
United Kingdom
gavin@krai.ai

Anton Lokhmotov
Krai Ltd
United Kingdom
anton@krai.ai

***Abstract*—We present how we solved the challenges we faced in achieving linear scaling for MLPerf™ Inference vision benchmarks on Datacenter and Edge servers equipped with Qualcomm Cloud AI 100 accelerators.**

The MLPerf Inference benchmarks for Computer Vision include: one Image Classification benchmark – ResNet50, and two Object Detection benchmarks – SSD-ResNet34 and SSD-MobileNet-v1, each presenting its own performance scaling challenges.

The server configurations include: one GIGABYTE server with 128 AMD EPYC™ physical cores and with 16 Qualcomm Cloud AI 100 cards; two GIGABYTE servers with 32 AMD EPYC™ physical cores and with 5 or 8 Qualcomm Cloud AI 100 cards.

Our results from the v1.1 submission round include the highest ResNet50 Server score and the highest SSD-MobileNet Offline score in the history of MLPerf.

***Index Terms*—performance, scaling, benchmarks, vision, machine learning, deep learning, inference, MLPerf, ResNet50, SSD-ResNet34, SSD-MobileNet, Qualcomm, Cloud AI 100, AMD, EPYC**

I. Introduction

MLCommons™ is a non-profit organization aiming to accelerate innovation in Machine Learning (ML). MLPerf Inference, the working group of MLCommons focused on benchmarking ML inference, received its first submissions in late 2019, with 4 rounds to late 2021: v0.5, v0.7, v1.0, v1.1. In this paper, we adhere to the rules [1] that were used in the v1.1 round.

A. Divisions

MLPerf Inference defines two divisions: Closed and Open. In this paper, we only consider submissions to the Closed division, where the strict rules allow apples-to-apples comparison between different hardware without such techniques as e.g. weights pruning (decreasing the number of operations) and quantization-aware training (increasing the accuracy).

B. Workloads

The MLPerf Inference benchmark suite [2] includes one Image Classification workload using the ResNet50-v1.5 model operating on the ImageNet 2012 validation dataset, and two Object Detection workloads using the SSD-ResNet34 and SSD-MobileNet-v1 models operating on the COCO 2017 validation dataset.

C. Categories

The Datacenter category covers large servers on-premises and in the cloud. The Edge category covers small servers on-premises and edge appliances.

ResNet50 and SSD-ResNet34 can be submitted under both the Datacenter and Edge categories. SSD-MobileNet can only be submitted under the Edge category.

D. LoadGen

MLPerf Inference uses an API called LoadGen to generate inference *queries* for the *system-under-test* (SUT) according to several *scenarios*. Benchmark implementors must integrate the LoadGen API and system-specific APIs for inference to create compliant and performant implementations.

E. Scenarios

All Closed Datacenter submissions must include the Offline and Server scenarios, while all Closed Edge submissions must include the Offline and Single Stream scenarios. In this paper, we only consider the Offline and Server scenarios.

In the Offline scenario, the SUT receives a single query containing all samples for inference; in other words, all samples are available at once. In the Server scenario, the SUT receives queries with one sample per query; the queries are fired according to a Poisson distribution based on the expected number of queries per second (QPS) that the SUT can handle.

F. Input size

The vision models take as input square images with 3 channels (RGB), as shown in Table I. If the weights of a model are represented as 32-bit floating-point numbers (`float`, in C parlance), the input size in bytes is obtained by multiplying the number of pixels by 3 (the number of channels), and then by 4 (the number of bytes per 32-bit number). If the weights of a model are represented as 8-bit integer numbers (`char` or `int8`, in C parlance), the input size in bytes is obtained by multiplying the number of pixels by 3 (the number of channels).

TABLE I: Workload Characteristics

Benchmark	Validation Dataset	Input Size *(pixels)*	Minimum Buffer Size *(the number of images)*	Server Latency Constraint *(milliseconds)*
ResNet50	ImageNet 2012	224×224	1024	15
SSD-ResNet34	COCO 2017	1200×1200	64	100
SSD-MobileNet	COCO 2017	300×300	256	N/A (Edge only)

G. Buffer size

To eliminate any effects of caching, the rules specify the minimum buffer size in terms of the number of samples that must be loaded into main memory when measuring performance. This buffer size ("performance sample count") is also given in Table I. For 8-bit quantized models, the corresponding minimum buffer sizes in bytes is:

- for ResNet50: $1024 \times 224 \times 224 \times 3 \approx 154$ MB;
- for SSD-ResNet34: $64 \times 1200 \times 1200 \times 3 \approx 276$ MB;
- for SSD-MobileNet: $256 \times 300 \times 300 \times 3 \approx 69$ MB.

H. Server Latency Constraints

In the Server scenario, 99% of the queries must be processed within the given, benchmark-specific time; in other words, the 99%-percentile *latency* must be lower than the given time. These *latency constraints* are given in Table I for ResNet50 and SSD-ResNet34. If this constraint is not satisfied (exceeded), the result gets marked as `INVALID`.

The Offline scenario does not have an associated latency constraint.

II. Performance Scaling

Since the v1.0 round, we have been engaged in implementing, validating and optimizing MLPerf Inference benchmarks for the Qualcomm Cloud AI 100 architecture for inference acceleration. This product line includes Half-Height Half-Length (HHHL) PCIe cards having 75 Watt TDP and Dual M.2 (DM.2) modules having 15–25 Watt TDP. In this paper, we mostly consider PCIe cards for servers.

Table II gives the Offline performance of a single PCIe card for each benchmark, along with the number of samples hardware processes in parallel ("batch size"). For Qualcomm Cloud AI 100, the optimal batch size is typically a single-digit number, unlike for GPUs where the optimal batch size is typically a 3–4 digit number.

TABLE II: Performance of a single Qualcomm Cloud AI 100 card under the Offline scenario (with SDK v1.5.6.).

Benchmark	Samples per second	Batch Size
ResNet50	22667	8
SSD-ResNet34	435	1
SSD-MobileNet	19363	4

For our experiments, we used 3 GIGABYTE servers with AMD EPYC processors and Qualcomm Cloud AI 100 cards, as specified in Table III. One R282-Z93 server was equipped with 5 cards and used for Edge submissions. The other R282-Z93 server and the G292-Z43 server were equipped with 8 and 16 cards, respectively, and were used for Datacenter submissions. All the servers were running under the CentOS 7.9 OS, with the Linux kernel 5.4.1.

In this section, we describe challenges for achieving linear scaling, and techniques we used to overcome them.

A. Offline Preprocessing

The ImageNet and COCO datasets contain JPEG images of various sizes, which need to be scaled to the input dimensions of the workloads (Table I). As permitted by the MLPerf Inference rules, input preprocessing such as decoding and slacing can be performed offline, i.e. before performance measurements. Since the models we used for submission were quantized to `int8` for performance reasons, we also preprocessed the input images from `float` to `int8` per pixel.

Offline preprocessing not only reduces the size of data stored on disk (from 30.1 GB to 7.5 GB for ResNet50, from 86.4 GB to 21.6 GB for SSD-ResNet34, and from 5.4 GB to 1.35 GB for SSD-MobileNet), but also the size of data that we need to copy from main memory to accelerator memory at run-time.

B. Using Fast Memory Copy

Even with the reduction in data size, copying data can become a bottleneck. Unfortunately, the `memcpy` routine in GLIBC 2.17 (which comes with CentOS 7), is not vectorized.

We improved the speed of data copy by using a 256-bit vector (AVX2) implementation.

C. Avoiding the DDR Bottleneck

LoadGen (§I-D) dispatches queries from a fixed-size memory buffer (§I-G). Sharing this memory buffer between multiple cards can result to a bottleneck due to insufficient memory bandwidth. For example, a single card performing ResNet50 inference at the rate of ≈ 22000 samples per second, needs to read $\approx 22000 \times 224 \times 224 \times 3 \approx 3.2$ GB per second. A DDR4 module operating at 3.2 GHz has the peak bandwidth of 25.6 GB/s. Therefore, at $25.6/3.2 = 8$ cards, we hit the DDR limit if LoadGen requests are to be served through a single DDR memory channel.

To solve this problem, we can replicate the LoadGen buffer. For simplicity, we replicate the buffer once per card since the buffer is only a few hundred MB in size (§I-G).

D. Affining cards to CPU sockets

On the R282, the per-card performance remains practically the same whether we use a single card or all 5 or 8 cards together. But on the G292, there is up to a 50% reduction in per-card performance when all 16 cards are used.

Server	CPU	# Cards	# Physical Cores	RAM	NUMA	Submission Category
GIGABYTE R282-Z93	AMD EPYC 7282 (Rome)	5	32	512 GB	NPS1	Edge Server
GIGABYTE R282-Z93	AMD EPYC 7282 (Rome)	8	32	512 GB	NPS1	Datacenter
GIGABYTE G292-Z43	AMD EPYC 7713 (Milan)	16	128	1024 GB	NPS1	Datacenter

TABLE III: Server Configurations.

We found that this performance drop is due to a bottleneck in the memory path from the PCIe to the DRAM. By ensuring that the DRAM used by a card is indeed local to the physical socket, we could scale linearly to all 16 cards.

E. Managing the Threads

Since the latency is critical in the Server scenario (I), we had to ensure that the worker threads were not congested. We had two types of threads:

- card threads that mainly copy the data in/out from the cards; and
- host threads that copy the data in/out from the host memory to DMA memory of the cards.

To minimize the latency, we partitioned the physical CPU cores between the Qualcomm Cloud AI 100 cards, ensuring the local socket condition (§II-D), as well as the best L3 cache utilization.

For the AMD Milan CPUs, the L3 cache is shared by every consecutive 8 physical CPU cores starting from core 0. To ensure the best cache availability to the CPU threads, we allocated the physical CPU cores to the cards as shown in Table IV. Here, for the G292 there is no sharing of L3 cache among any 2 cards; for the R282 having 8 cards, every 2 cards share the same L3 cache; for the R282 having 5 cards, we only affine the first 4 cards and let the fifth card use the entire system. Since the 5 card system is submitted in Edge category, we have no Server scenario for this and hence no latency constraint too.

F. Running NMS on the Host CPU

The Non-Maximum Suppression (NMS) operation is used to filter out predictions of SSD-based models. Since this operation involves control flow, it is better suited to run on the CPU. Therefore, we cut off the NMS operation from the two Object Detection models and optimized its CPU implementation. By pipelining the output from the cards to the CPU, we could achieve practically the same workload throughput with and without the NMS operation.

For the SSD-ResNet34 workload, running NMS on the host side means that instead of getting a few kilobytes of output per image we are now getting $15130 \times 81 \times 2 + 15130 \times 4 \approx 2.5$ MB of output per image, which needs to be transferred from the card to the host. Here, 15130 is the number of bounding boxes for an image and 81 is the number of object classes being considered for SSD-ResNet34 and we are quantizing the confidence score to `fp16` (2 bytes), whereas the 4 bounding box coordinates are quantized to `uint8` (1 byte). At 435 samples per second, this means an output side data transfer of 435×2.5 MB ≈ 1 GB/s which though significant is not a bottleneck. Also, at the rate of 435 samples per second, NMS should not become a bottleneck if it can finish processing in $1/435 \approx 2.3$ ms.

For the SSD-MobileNet workload, running NMS on the host side was even more trickier. Here, we have 1917 bounding boxes and 90 object classes. Both the confidence score and bounding box coordinates are quantized to `uint8`. So, for each image the output side data transfer becomes $1917 \times 90 + 1917 \times 4 \approx 180.2$ KB and at 19363 samples per second this means a data transfer rate of ≈ 3.5 GB/s which is 3.5 times that of SSD-ResNet34. Also, the latency of NMS operation per image needed to be below $1/19363 \approx 51.6\mu s$ to avoid NMS becoming a bottleneck.

The layout of the tensor representing the confidence score to the NMS operation is different for SSD-ResNet34 and SSD-MobileNet. For SSD-ResNet34, the outer dimension is the image classes and the inner dimension is the bounding boxes; this is reverse for SSD-MobileNet. So provided a different implementation for each model avoiding the need to transpose the tensor. Our implementation is available at NMS-ABP [1].

III. Results

A. Performance Scaling

The performance results of our benchmark implementations are given in Figure 1. The performance of a single card (Table II) scales almost linearly to 5, 8 and 16 cards for ResNet50 and SSD-ResNet34. For SSD-MobileNet, we get linear scaling for 5 cards. (Since SSD-MobileNet is an Edge only benchmark, we do not run it on 8 and 16 cards.)

We now compare the performance of Qualcomm Cloud AI 100 powered submissions to other submissions to MLPerf Inference v1.1.

B. Datacenter Category

Table V give details of each submission ID mentioned in the result figures for the Datacenter category, while Table VI does the same for the Edge category.

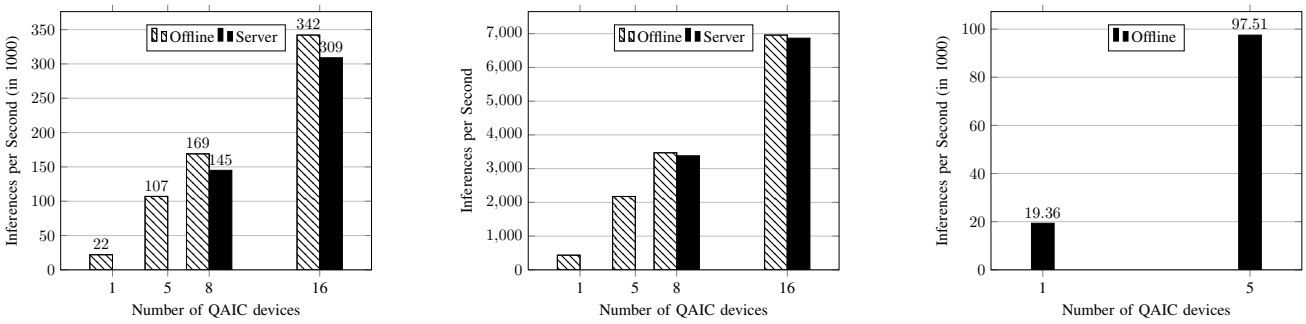
For comparison purposes, we have taken the top 8 submissions to the Closed division and the top 6 submissions to the Closed/Power division. Since Qualcomm's Submission 056 is the only entry among the top 8 submissions with a power figure, we thus have a total of 13 unique entries in this table. Interestingly, only Qualcomm accelerators and NVIDIA GPUs are used in these submissions. For full submission details please refer to MLPerf Inference v1.1 - Datacenter. [2]

[1] https://github.com/krai/ck-mlperf/tree/master/package/lib-nms-abp

[2] https://mlcommons.org/en/inference-datacenter-11/

System	# Cards	Card ID	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
R282-Z93	5	Core	0-7	8-15	16-23	24-31	0-31											
R282-Z93	8	Mapping	0-3	4-7	8-11	12-15	16-19	20-23	24-27	28-31								
G292-Z43	16		64-71	72-79	80-87	88 -95	96-103	104-111	112-119	120-127	0-7	8-15	16-23	24-31	32-39	40-47	48-55	56-63

TABLE IV: CPU affinity.



(a) ResNet50 (b) SSD-ResNet34 (c) SSD-MobileNet

Fig. 1: Performance Scaling

Sub. ID	Submitter	System	Processor	#	Accelerator	#	Power
056	Qualcomm	GIGABYTE G292-Z43	AMD EPYC 7713	2	Qualcomm Cloud AI 100	16	Y
001	Dell	Dell EMC DSS 8440	Intel(R) Xeon(R) Gold 6248R	2	NVIDIA A100-PCIE-80GB	10	N
021	Inspur	Inspur NF5488A5	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	8	N
022	Inspur	Inspur NF5688M6	Intel(R) Xeon(R) Platinum 8358	2	NVIDIA A100-SXM-80GB	8	N
064	Supermicro	Supermicro SYS-420GP-TNR	Intel(R) Xeon(R) Platinum 8360Y	2	NVIDIA A100-PCIe-40GB	10	N
047	NVIDIA	NVIDIA DGX A100	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	8	N
049	NVIDIA	NVIDIA DGX A100	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	8	N
034	NVIDIA	GIGABYTE G482-Z54	AMD EPYC 7742	2	NVIDIA A100-PCIe-80GB	8	N
048	NVIDIA	NVIDIA DGX A100 (MaxQ)	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	8	Y
037	NVIDIA	GIGABYTE G482-Z54 (MaxQ)	AMD EPYC 7742	2	NVIDIA A100-PCIe-40GB	8	Y
058	Qualcomm	GIGABYTE R282-Z93	AMD EPYC 7282	2	Qualcomm Cloud AI 100	8	Y
016	Dell	Dell EMC PowerEdge XE8545	AMD EPYC 7763	2	NVIDIA A100-SXM-80GB	4	Y
006	Dell	Dell EMC PowerEdge R750xa (MaxQ)	Intel(R) Xeon(R) Platinum 8368	2	NVIDIA A100-PCIE-40GB	4	Y

TABLE V: Top submissions to MLPerf Inference v1.1 - Datacenter.

1) Performance per Socket: As shown in Table V, all the top submissions in the Datacenter category are done on dual-socket systems. Figure 2 thus shows the top 8 submissions as well as the top 8 submissions on dual-socket systems.

For ResNet50, Qualcomm's Submission 056 using 16 cards wins both the Offline and Server scenario scores. Moreover, our low latency implementation ensured that even at this highest throughput we still have the best Server/Offline ratio of 90.6% among the top 8 submissions.

For SSD-ResNet34, Dell's Submission 001 using 10 NVIDIA A100 GPUs achieves the highest performance. Here, the peak NVIDIA number is 1.3 times higher than the peak Qualcomm number. In terms of the Server/Offline ratio, Qualcomm's Submission 056 again wins with a score of 98.7% versus the next best of 97.9% by Dell's Submission 001.

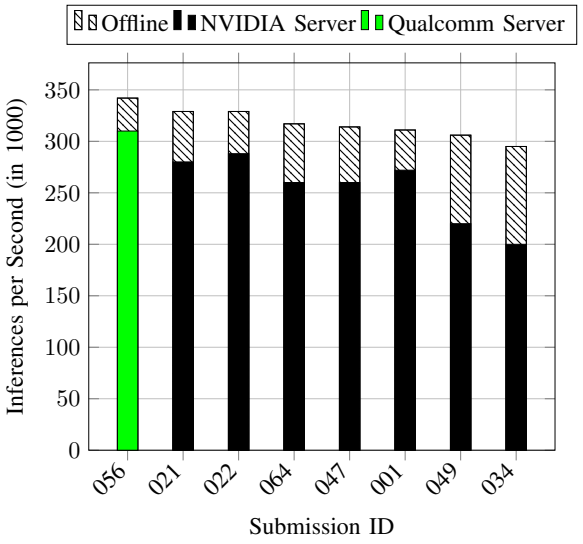
2) Performance per Watt: Figures 3 and 4 show the top 6 submissions with power measurements and their power efficiencies. The best Performance per Watt for both ResNet50 and SSD-ResNet34 is achieved by our Submission 058 on the R282-Z93 using 8 cards with the Performance per Watt scores of 197.4 and 4.03 for ResNet50 and SSD-ResNet34, respectively. Our Submission 056 on the G292-Z43 using 16 cards is in the second place with the Performance per Watt scores of 185.4 and 3.58. The best Performance per Watt for the two benchmarks for any NVIDIA submission is 112.03 and 2.62, making Qualcomm a runaway winner in terms of energy efficiency by the factors of 1.76 and 1.54, respectively.

C. Edge Category

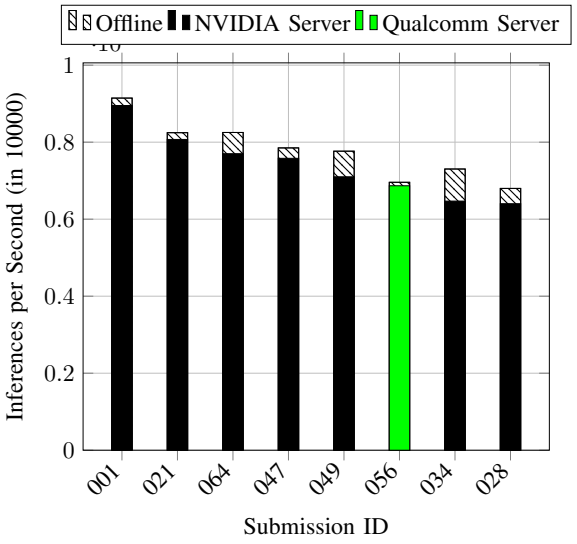
Table VI shows the top 8 submissions and the top 6 submissions with power measurements in the Edge category. 2 of the top 8 submissions are with power measurements, and so we have a total of 12 entries in this table. For full submission details please refer to MLPerf Inference v1.1 - Edge [3]. Compared with the Datacenter category, the Edge category has one additional vision workload: Object Detection (small) using the SSD-MobileNet model.

1) Performance per Socket: Unlike the top submissions in the Datacenter category where all the top submissions are done on dual-socket systems, here we have 4 submissions on single-socket systems. But all the single socket submissions are aimed at power efficiency, thus not competing in terms of the peak performance. So, all the top 8 performance numbers in the Edge category are on dual socket systems still, as shown in Figure 5.

[3]https://mlcommons.org/en/inference-edge-11/



(a) ResNet50

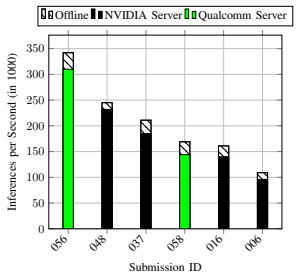


(b) SSD-ResNet34

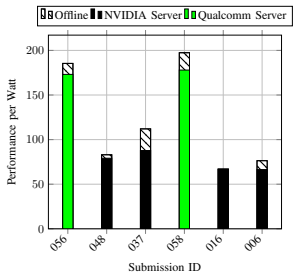
Fig. 2: Top submissions in the Datacenter category.

Sub. ID	Submitter	System	Processor	#	Accelerator	#	Power
124	Qualcomm	GIGABYTE R282-Z93	AMD EPYC 7282	2	QUALCOMM Cloud AI 100 PCIe HHHL	5	Y
077	Inspur	NE5260M5	Intel(R) Xeon(R) Gold 6258R	2	NVIDIA A100-PCIe	2	N
079	Inspur	NE5260M5	Intel(R) Xeon(R) Gold 6258R	2	NVIDIA A100-PCIe	2	N
078	Inspur	NE5260M5	Intel(R) Xeon(R) Gold 6258R	2	NVIDIA A100-PCIe	2	Y
083	Inspur	NF5688M6	Intel(R) Xeon(R) Platinum 8358	2	NVIDIA A100-SXM-80GB	1	N
081	Inspur	NF5488A5	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	1	N
115	NVIDIA	NVIDIA DGX A100	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	1	N
116	NVIDIA	NVIDIA DGX A100	AMD EPYC 7742	2	NVIDIA A100-SXM-80GB	1	N
075	Dell	PowerEdge XE2420 (MaxQ)	Intel(R) Xeon(R) Gold 6252N	2	NVIDIA A10	1	Y
121	Qualcomm	Edge AI Development Kit	Qualcomm Snapdragon 865	1	QUALCOMM Cloud AI 100 DM.2	1	Y
120	Qualcomm	Edge AI Development Kit	Qualcomm Snapdragon 865	1	QUALCOMM Cloud AI 100 DM.2e	1	Y
111	NVIDIA	Auvidea X220-LC (MaxQ)	NVIDIA Carmel (ARMv8.2)	1	NVIDIA AGX Xavier 32GB	1	Y

TABLE VI: Top submissions to MLPerf Inference v1.1 - Edge.

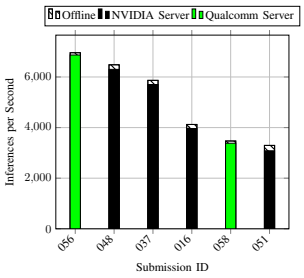


(a) Performance

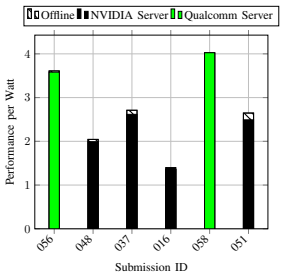


(b) Performance per Watt

Fig. 3: Performance and power efficiency on the ResNet50 workload in the Datacenter category.



(a) Performance



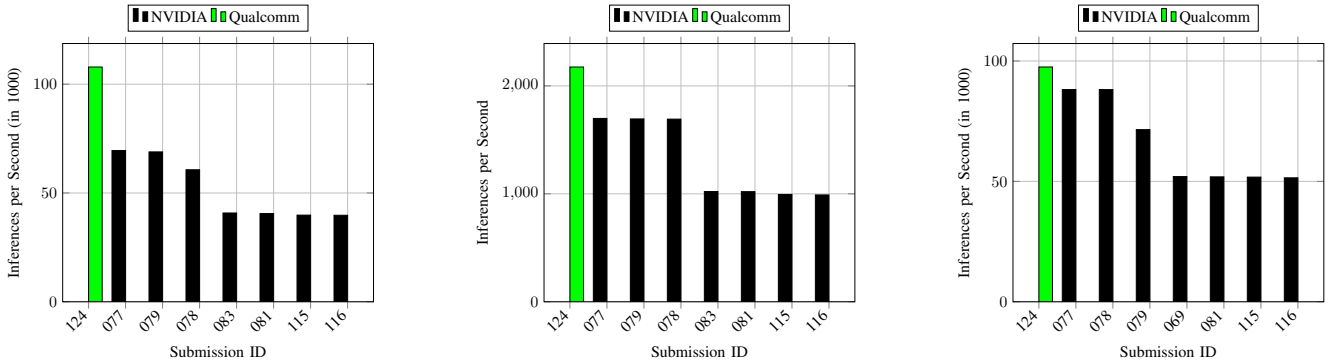
(b) Performance per Watt

Fig. 4: Performance and power efficiency on the SSD-ResNet34 workload in the Datacenter category.

In the Edge category, Qualcomm's Submission 124 using 5 cards achieves the peak performance for all the three vision benchmarks. The relative performance difference between the Qualcomm submission and the next best submissions using NVIDIA GPUs, for ResNet50, SSD-ResNet34 and SSD-MobileNet is 55%, 28% and 6%, respectively.

2) Performance per Watt: Figures 6, 7 and 8 show the top 6 Edge submissions with power measurements and their power efficiencies. The peak performing Submission 124 for per socket performance for all the three benchmarks is submitted with power measurements and hence tops here too.

The best Performance per Watt for all the three benchmarks

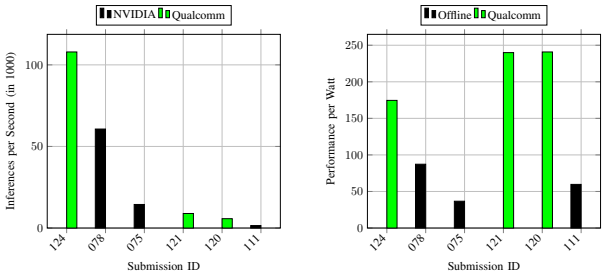


(a) ResNet50 (b) SSD-ResNet34 (c) SSD-MobileNet

Fig. 5: Top submissions in the Edge category.

is achieved by Submission 121 on the Qualcomm Edge AI Development Kit under the 20W TDP constraints. A close second place is achieved by Submission 120, again on the Qualcomm Edge AI Development Kit but under the 15W TDP constraints. Both submissions use the same Qualcomm Cloud AI 100 architecture and toolchain as the server-based submissions (which are the focus of this paper) but configured differently than server cards under the 75W TDP constraints. The benchmark implementation is also the same but differently configured.

The best Performance per Watt figures in the Edge category achieved by Qualcomm for ResNet50, SSD-ResNet34 and SSD-MobileNet are 239.9, 4.82 and 141.7 respectively. These figures are better than the best NVIDIA submissions by the factors of 2.74, 2.99 and 1.8, respectively.



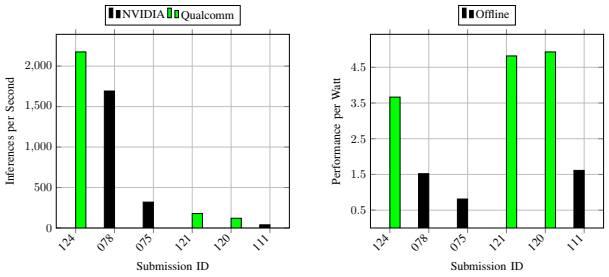
(a) Offline Performance (b) Performance per Watt

Fig. 6: Performance and power efficiency on the ResNet50 workload in the Edge category.

IV. Conclusion

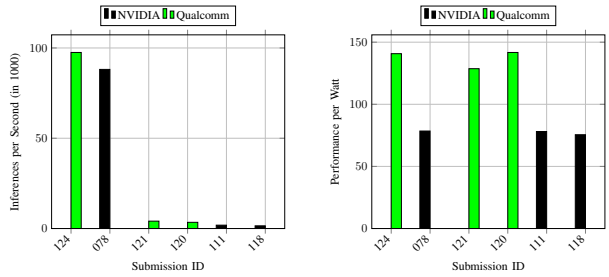
We have presented a summary of our submissions the MLPerf Inference v1.1 round using Qualcomm Cloud AI 100 accelerators. We have also presented performance comparisons between top Vision benchmark submissions to the Closed division for both the Datacenter and Edge categories. All the top submissions are achieved with either Qualcomm accelerators or NVIDIA GPUs.

In the Datacenter category, Qualcomm has the peak performance score for ResNet50, while NVIDIA has the same for



(a) Offline Performance (b) Performance per Watt

Fig. 7: Performance and power efficiency on the SSD-ResNet34 workload in the Edge category.



(a) Offline Performance (b) Performance per Watt

Fig. 8: Performance and power efficiency on the SSD-MobileNet workload in the Edge category.

SSD-ResNet34. In terms of energy efficiency, the Qualcomm submissions outperform the best NVIDIA submissions by the factors of 1.76 and 1.54 for ResNet50 and SSD-ResNet34, respectively.

In the Edge category, Qualcomm has the peak performance scores for all the three vision benchmarks. In terms of energy efficiency, the Qualcomm submissions outperform the best NVIDIA submissions by the factors of 2.74, 2.99 and 1.8 for ResNet50, SSD-ResNet34 and SSD-MobileNet, respectively.

References

[1] The MLCommons Association. MLPerf Inference rules. https://github.com/mlcommons/inference_policies/blob/master/inference_rules.adoc,

2018–2022.

[2] Vijay Janapa Reddi, Christine Cheng, David Kanter, Peter Mattson, Guenther Schmuelling, Carole-Jean Wu, Brian Anderson, Maximilien Breughe, Mark Charlebois, William Chou, Ramesh Chukka, Cody Coleman, Sam Davis, Pan Deng, Greg Diamos, Jared Duke, Dave Fick, J. Scott Gardner, Itay Hubara, Sachin Idgunji, Thomas B. Jablin, Jeff Jiao, Tom St. John, Pankaj Kanwar, David Lee, Jeffery Liao, Anton Lokhmotov, Francisco Massa, Peng Meng, Paulius Micikevicius, Colin Osborne, Gennady Pekhimenko, Arun Tejusve Raghunath Rajan, Dilip Sequeira, Ashish Sirasao, Fei Sun, Hanlin Tang, Michael Thomson, Frank Wei, Ephrem Wu, Lingjie Xu, Koichi Yamada, Bing Yu, George Yuan, Aaron Zhong, Peizhao Zhang, and Yuchen Zhou. MLPerf Inference Benchmark. In *Proceedings of the ACM/IEEE 47th Annual International Symposium on Computer Architecture*, ISCA '20, page 446–459. IEEE Press, 2020.