High-Speed, Real-Time, Spike-Based Object Tracking and Path Prediction on Google Edge TPU

Jonah Sengupta[1], Rajkumar Kubendran[2], Emre Neftci[3] and Andreas Andreou[1]
[1]John Hopkins University, Baltimore, MD, USA, [2]University of California San Diego, La Jolla, CA, USA,
[3]University of California Irvine, Irvine, CA, USA

***Abstract*—As computation and advanced high-dimensional signal processing is pushed to edge computational devices, energy efficient, unconventional architectures are needed to ameliorate this growing need. The Google Edge TPU, first used on a Cloud platform, is one such accelerator that is now commercially available for consumer use. Similarly, low-power, data-efficient vision sensors, such as the Dynamic Vision Sensor (DVS), have been developed and commercialized as well to improve upon the large data redundancy seen in these ML applications. This live demonstration is linking these two technologies to benchmark the Coral Edge TPU Board in a high-speed object tracking and prediction application. In comparison to a floating point architecture of similar form factor, the Intel Compute Stick, the Edge TPU has been show to outperform in terms of latency and computational efficiency.**

***Index Terms*—Edge Computing, Tensor Processing unit (TPU), Dynamic Vision Sensor (DVS).**

I. Introduction

Due to the rise of edge computing for applications in artificial intelligence and machine learning, novel architectures have been fabricated to handle these workloads in a time and energy efficient manner. One such architecture is the Google TPU, first used in server platforms to accelerate remote inference, but now commercially available as a USB companion or as a self-contained development board, the Coral Edge TPU Dev Board [1]. At the Telluride Neuromorphic Workshop, the Coral Edge TPU Dev Board was used to template real-time, high-speed object tracking and path prediction. Specifically, the accelerator hardware was paired with a spike-based, vision sensor, the DAVIS240C [2], which performed inference only using asynchronous, event-based data. This is similar to tracking tasks performed by other energy efficient, edge computational devices [3], [4].

II. Demo Setup

The live demonstration of this platform comprised of two distinct parts each performing two distinct tasks. This system was designed and trained using spiking data capturing a ball moving around an enclosure and bouncing off the boundaries. Using this data, the first task implemented was a simple tracker was trained using TensorFlow to track the ball. Given a bundle of spikes from the event-based sensor, the neural network will output an inferred x,y pixel location of the center of the object. Secondly, these inferred pixel locations were used to linearly estimate a velocity and predict the path of the ball 50 ms into the future via a trained multi-layer perceptron network. To realize a real-time demonstration, these two tasks were implemented on two different compact, computational platforms: the Intel Compute Stick, a full 64-bit, floating point architecture and the aforementioned Google Edge TPU Dev, a 8-bit, fixed point architecture. The general set-up of the demonstration comprises of two DAVIS240 sensors viewing a bouncing ball simulator. The spikes from the sensors are sent via USB to an Edge TPU and the Intel Compute Stick. The tracking Edge TPU communicates with a second TPU who performs the path prediction inference via UDP. Finally, the second TPU send the tracked point and prediction to the Intel Compute Stick, who also is performing inference as well, to display the results.

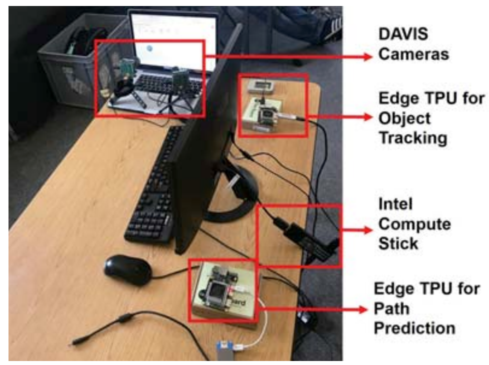

Fig. 1. Live Demo Setup for Ball Tracking and Path Prediction with Google Edge TPU, DAVIS cameras and Intel Compute Stick.

TABLE I
PERFORMANCE COMPARISON

Metric	Intel Compute Stick	Google Edge TPU
Power (W)	5.48	1.72
Model Size (MB)	1.8	0.5
Latency (ms)	27	4
Comp. Eff. ($\frac{GOps}{mW}$)	0.052	1.057

III. Conclusion

Ultimately, the two platforms have been benchmarked with these small networks. The 64-bit architecture consumed 5.5W of power while taking 27ms to perform the inference resulting in an efficiency of 52 MOPs/W. In contrast, the Edge TPU only consumed 1.72W of power while performing the inference in 4ms yielding a computational efficiency of 1.052 GOPs/W which are summarized in Table I.

REFERENCES

[1] "Coral edge tpu dev board." [Online]. Available: https://coral.ai/docs/dev-board/datasheet/

[2] C. Brandli, R. Berner, M. Yang, S.-C. Liu, and T. Delbruck, "A 240$\times$ 180 130 db 3 μs latency global shutter spatiotemporal vision sensor," *IEEE Journal of Solid-State Circuits*, vol. 49, no. 10, pp. 2333–2341, 2014.

[3] M. Evanusa, Y. Sandamirskaya *et al.*, "Event-based attention and tracking on neuromorphic hardware," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019, pp. 0–0.

[4] A. Amir, B. Taba, D. Berg, T. Melano, J. McKinstry, C. Di Nolfo, T. Nayak, A. Andreopoulos, G. Garreau, M. Mendoza *et al.*, "A low power, fully event-based gesture recognition system," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 7243–7252.