NASA SpaceCube Edge TPU SmallSat Card for Autonomous Operations and Onboard Science-Data Analysis

Justin Goodwill, Gary Crum, James MacKinnon, Cody Brewer,
Michael Monaghan, Travis Wise, Christopher Wilson
NASA Goddard Space Flight Center
8800 Greenbelt Rd, Greenbelt, MD, 20771; 301-286-1417
justin.goodwill@nasa.gov

ABSTRACT

Using state-of-the-art artificial intelligence (AI) frameworks onboard spacecraft is challenging because common spacecraft processors cannot provide comparable performance to datacenters with server-grade CPUs and GPUs available for terrestrial applications and advanced deep-learning networks. This limitation makes small, low-power AI microchip architectures, such as the Google Coral Edge Tensor Processing Unit (TPU), attractive for space missions where the application-specific design enables both high-performance and power-efficient computing for AI applications. To address these challenging considerations for space deployment, this research introduces the design and capabilities of a CubeSat-sized Edge TPU-based co-processor card, known as the SpaceCube Low-power Edge Artificial Intelligence Resilient Node (SC-LEARN). This design conforms to NASA's CubeSat Card Specification (CS2) for integration into next-generation SmallSat and CubeSat systems. This paper describes the overarching architecture and design of the SC-LEARN, as well as, the supporting test card designed for rapid prototyping and evaluation. The SC-LEARN was developed with three operational modes: (1) a high-performance parallel-processing mode, (2) a fault-tolerant mode for onboard resilience, and (3) a power-saving mode with cold spares. Importantly, this research also elaborates on both training and quantization of TensorFlow models for the SC-LEARN for use onboard with representative, open-source datasets. Lastly, we describe future research plans, including radiation-beam testing and flight demonstration.

I. INTRODUCTION

One of the fastest growing ground-based areas of research is artificial intelligence (AI), which has revolutionized a variety of application domains. Consequently, substantial commercial investment of applied AI is demonstrated through autonomous cars (e.g. Waymo, General Motors, Mercedes), social "bots", virtual assistants (e.g. Siri, Cortana, Alexa, Bixby), and strategic game systems (e.g. Watson, AlphaGo). Additionally, developers seek to integrate more AI into broader customer bases with smaller, more power efficient, AI microchips and accelerators, specifically targeting mobile and embedded markets. These advances in AI algorithms and custom accelerator electronics can also be harnessed to enable numerous breakthrough capabilities in the space domain, including autonomous swarm/constellation management, reactive health and status monitoring, and responsive, onboard data analysis. Furthermore, it is advantageous to combine next-generation, high-performance computing together with onboard intelligent co-processing. This synergistic combination is highly valuable because it would enable dynamic, onboard programming and reconfiguration of the intelligent co-processor allowing the device to change functions and applications in real time. For example, this feature could allow the system to rapidly change functions for different scenarios, which is necessary because each type of event may require a specific AI model to be programmed or swapped out on the device. This capability would enable the system to respond to different situations or objectives, such as switching from disaster detection mode (e.g., earthquakes, tsunamis, floods, and fires), to highly accurate targeting modes (e.g., specific object-targeting data).

While there is exciting potential and many benefits for deploying advanced AI applications in space, using commercial AI frameworks onboard spacecraft is challenging because traditional radiation-hardened (rad-hard) processors and other common spacecraft processors cannot provide the necessary onboard computing resources and processing capability to effectively deploy complex AI models. Therefore, they would be substantially restricted to simpler machine-learning approaches. This limitation makes small, low-power AI microchip architectures, such as Tensor Processing Units (TPUs), attractive for space missions where the application-specific design enables both high-performance and power-efficient computing for AI applications.

To address these design considerations, this research enables the use of state-of-the-art, experimental, AI

microchip architectures (specifically the Google Coral Edge TPU [1]) for SmallSat platforms. In this paper, we introduce the design and capabilities of a CubeSat-sized Edge TPU-based processor card, known as the SpaceCube Low-power Edge Artificial Intelligence Resilient Node (SC-LEARN), built to NASA's CubeSat Card Specification (CS2) [2] for integration into SmallSat systems. The SC-LEARN features a configurable system with three Edge TPUs. The supporting circuitry and components around the Edge TPUs are reliable, space-qualified components, and built to NASA standards. This card is designed to be monitored by a complementary high-performance processor, which is responsible for powering on/off the individual Edge TPU modules.

The approach for this design was to create a CubeSat-sized 1U interface card to integrate into NASA Goddard's reliable CubeSat architecture (Modular Architecture for a Resilient Extensible SmallSat - MARES [3]) and initially target the Edge TPU. The primary benefits of the design were: (1) the Google Coral Edge TPU has several advantages over its close competitors, (2) the compatibility of this design with the Goddard CubeSat architecture provides reliable operation and monitoring of the card, and (3) the studies into the supporting software ecosystem will allow for onboard programming and reconfiguration of the AI microchip.

II. BACKGROUND

The following sections describe the current state-of-the-art for onboard AI-enabled devices and the Edge TPU processing device. Additionally, we briefly describe the CubeSat form-factor the SC-LEARN conforms to and the high-performance space processor the SC-LEARN operates with cooperatively. Finally, this section describes the test datasets, TensorFlow models, and mission use-case for the SC-LEARN.

AI For Science and Defense

The impending need for specialized low-power AI chips to enable advanced onboard capability has been heavily emphasized in both science and defense applications. While the science and defense domains have differing application goals, general-purpose AI microchips, such as the Edge TPU, can be an enabling technology for a broad variety of scenarios.

For science, intelligent and autonomous systems have been emphasized in numerous guiding NASA documents. These documents include NASA's 2017 Strategic Technology Investment Plan [4] and the 2015 NASA Technology Roadmaps [5], which specifically highlight "robotics and autonomous systems" as a critical technology investment and describe eleven technology areas where autonomy and artificial intelligence can provide enhanced capability. Furthermore, the significance of AI research is subsequently elaborated in the new NASA Technology Taxonomy 2020 [6], where the 2020 update specifically identifies and addresses advances in AI. The research presented here is directly applicable to TX05.5.1 with machine learning and artificial intelligence in cognitive networking, TX10.1 for situational and self-awareness, and most significantly, TX11.4.2 which focuses on intelligent data understanding for automatic analysis of datasets.

These technology focus areas are applied in the Earth science decadal survey, *Thriving on Our Changing Planet* [7], towards the highly valuable automatic classification of vegetation and natural phenomena using spectral remote sensing. This application is one of many ideal candidates for the proposed Edge TPU design. The need for extremely low-power, specialized AI chips is not only described in Earth science but also planetary science. In *Visions into Voyages*, the planetary science decadal survey [8], the key capabilities identified are system autonomy and autonomous precision landing technology that represent two application domains where these types of AI microchips can be specifically fine-tuned to enable power-efficient solutions. Additional supporting use cases are prominently emphasized for Mars exploration (Emerging Capabilities for Mars Exploration [9]) explicitly citing limitations for onboard processing that can be accelerated with AI co-processing systems.

The need for AI in the defense domain spans across multiple agencies. The National Geospatial-Intelligence Agency emphasized onboard analysis to address massive data volume constraints in [10]. The "Air Force Space Command Long-Term Science and Technology Challenges" [11] memorandum specifically highlighted the need for automated and autonomous systems, artificial intelligence, and advanced computer architectures. However, the relevance of incorporating AI techniques into space applications is most profoundly illustrated in the Defense Advanced Research Projects Agency (DARPA) Blackjack program initiative. In their 2019 broad agency announcement [12], DARPA describes a processing system that will provide mission-level autonomy, classification, and high-performance computing. A core component of the Blackjack program is the "Pit Boss" edge processor, a payload processor unit assigned to autonomously task, collect, process, exploit, and disseminate multi-sensor data and/or signals in multiple warfighter domains. Finally, [13] provides an integrated government-wide strategy for AI-accelerated conflict.

Related Works

Machine learning and onboard analysis have been strategically important and applied by NASA from as early as EO-1 (Earth Observing-1) [14], the large observation satellite. EO-1 explored autonomy using a rad-hard Mongoose-V. Some years later, IPEX [15] demonstrated onboard classification with an Atmel AT91SAM9. However, more recent AI applications have been theorized and considered for calibration of sensors in spacecraft buses (e.g. calibration of magnetometers [16]), and for processing of onboard image data (e.g. cloud screening [17]). [18] describes CNN-based object detection using a NVIDIA Jetson Nano. Finally, [19] focused on swarms of SmallSats for in-space manufacturing, commenting on the benefits of machine learning for their use case. These earlier demonstrations used general-purpose CPUs or proposed studies on devices that could potentially be included within CubeSat-size restrictions or could benefit specific application domains. However, while AI embedded microchips are currently the cutting-edge embedded solutions, there are few examples of onboard deployment. One recent example is European Space Agency (ESA) miniaturized Visible-Near-InfraRed (VNIR) hyperspectral imager (HSI), called HyperScout-2, which uses the Intel Movidius Myriad 2 Vision Processing Unit (VPU) [20]-[21], as part of the ESA PhiSat-1 initiative. Another example is the University of Hawaii-led CubeSat Hyperspectral Thermal Imager (HYTI) mission, using another device in the Intel Movidius family, the Myriad X [24].

Google Coral Edge TPU

The Edge TPU, developed by Google Research, is a small, low-power ASIC designed to provide high-performance neural-net inferencing. The Edge TPU is a flexible design that supports general-purpose AI applications using the open-source TensorFlow Lite API, making it widely accessible for application development. Additionally, the Edge TPU device is based on extensively studied systolic-array architectures, making it broadly configurable for many AI applications. Unfortunately, the radiation characterization of the device is largely still unknown (although preliminary reports suggest promising characteristics); however, this limitation can be mitigated with proper system design and monitoring onboard the spacecraft. In addition, unlike many commercial devices, the Edge TPU has an extended operating temperature range (− 20˚C to 70˚C) and includes built-in safety features such as automated frequency throttling at high temperatures, which is necessary for the survival of the device in a space environment.

CubeSat Card Specification (CS2)

The CubeSat Card Specification (CS2) was developed at the NASA Goddard Science Data Processing Branch to establish a common template such that all future CubeSat-sized cards would be compatible for system designs. The specification, originally outlined in [2], describes pinout configurations along with mechanical and electrical specifications targeting the 1U CubeSat form-factor. Compliance with this specification allows previously developed backplane and mechanical enclosure elements to be quickly extended for new mission applications. Currently, NASA has developed several compliant cards (including single-board computers, power cards, and I/O cards) allowing developers to mix-and-match cards within the catalog to build new systems for missions. SC-LEARN complies with this specification, which defines several major design characteristics (e.g. board keep-outs, connector definitions, etc…) demonstrated in Section III. Compatibility with CS2 allows SC-LEARN to be included in current and future proposed designs. An example three-card configuration for a small AI processing unit is shown in Figure 1. This box configuration features the SC-LEARN card, a host processor card (SpaceCube v3.0 Mini), and the Low-Voltage Power Converter (LVPC) card, connected with a backplane design.

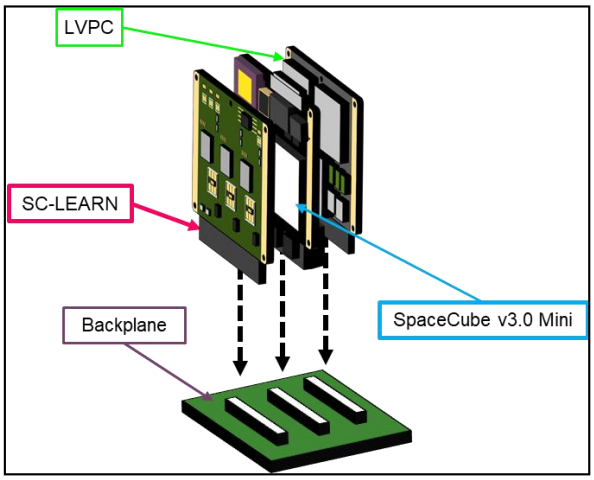

Figure 1: Three-Card AI Processing Box Configuration

SpaceCube Family of Processor Cards

SC-LEARN features several Edge TPU Accelerator Modules, however, these modules must be controlled and operated by a host processor. There are several processor cards in the CS2 form-factor that can be used cooperatively with SC-LEARN. The first is the SpaceCube v3.0 Mini [2], a 1U CubeSat-sized single-

Table 1: Dataset Parameter Summary

Dataset	Sensor	Spatial Dimensions	Spectral Bands	Sensitive Wavelengths	Classes	Labeled Pixels	GSD
Indian Pines	AVIRIS	145x145	224	0.4-2.5 μm	16	10,249	20 m
Salinas	AVIRIS	512x217	224	0.4-2.5 μm	16	54,129	3.7 m
Pavia University	ROSIS	610x610	103	0.43-0.85 μm	9	50,232	1.3 m

board computer that features the Xilinx Kintex UltraScale KU060 FPGA. The second is the SpaceCube Mini-Z, featuring a Xilinx Zynq-7020 device, which is a hybrid system-on-chip design combining a dual-core ARM Cortex-A9 processor with an Artix-7 FPGA fabric [2]. Finally, the last compatible card is the SpaceCube Mini-Z+, a further upgraded version of the SpaceCube Mini-Z equipped with more rad-hard power components, an upgraded rad-hard power sequencing circuit, an added rad-hard reset enable timeout circuit, and flight-grade oscillators and passives.

Datasets and Test Vectors

For early prototyping and demonstration of the Edge TPU's capabilities, several publicly available hyperspectral datasets were considered for study. Ultimately, three datasets were selected for use and described below: Indian Pines, Salinas, and Pavia University. These datasets were selected due to their variation in hyperspectral sensor types, wavelength ranges, ground sampling distances (GSDs), dataset sizes, and ground-truth classifications, as described in Table 1. Each dataset was additionally normalized on a per-band basis using Equation 1, where I represents one $h \times w$ band.

$$I_{norm} = \frac{I - \min(I)}{\max(I) - \min(I)} \quad (1)$$

Indian Pines [23]: This sample dataset was captured by the AVIRIS (Airborne Visible/Infrared Imaging Spectrometer) sensor (https://aviris.jpl.nasa.gov/) over agricultural fields in Northwestern Indiana. The dataset size is 145×145 pixels with 224 spectral bands, where 10,249 pixels are labeled according to 16 different classes. For this research, the bands corresponding to regions of water absorption were removed due to low signal-to-noise ratio (SNR), leaving 200 remaining bands for study.

Salinas Scene [23]: This sample dataset was captured by AVIRIS sensor over Salinas Valley, California. The dataset size is 512×217 pixels with 224 spectral bands. The dataset is also labeled with 16 different classes. Like the previous dataset, the bands corresponding to regions of water absorption were removed due to low SNR, leaving 200 remaining bands.

Pavia University [23]: This sample dataset was captured by the ROSIS (Reflective Optics System Imaging Spectrometer) sensor over Pavia in northern Italy. The Pavia University scene size is 610×610 pixels, which are labeled into 9 ground-truth classes. Like the two previous datasets, the bands corresponding to regions of water absorption were removed due to low SNR, leaving 100 bands for consideration.

STP-H9/SCENIC

The SC-LEARN is currently in development for inclusion in a multi-card, AI experiment demonstration on the International Space Station (ISS). The experiment, named SCENIC (SpaceCube Edge-Node Intelligent Collaboration), is a joint collaboration between NASA Goddard, the Aerospace Corporation, and the Air Force Research Laboratory Space Vehicles Directorate to demonstrate several processing units capable of supporting machine learning and artificial intelligence to perform a variety of science and defense imaging applications with a hyperspectral sensor. The Space Test Program (STP) [24] at the Department of Defense (DoD) is responsible for supporting the development, evaluation, and advancement of new technologies needed for the future of spaceflight. STP-Houston provides opportunities for both DoD and NASA to perform on-orbit research and technology demonstrations from the ISS. The SCENIC experiment has several key objectives:

- Demonstration and evaluation of commercial AI microchips (specifically the Intel Movidius Myriad X and Google Coral Edge TPU) for radiation characterization in a relevant space environment. The experiment additionally features the Xilinx Deep Learning Processor Unit (DPU) [25], an FPGA-based AI accelerator residing in the primary FPGA card, providing an FPGA-based AI option for comparison against the two AI microchips.
- Collection of an extensive HSI image archive of terrestrial scenes required to train data-driven deep

neural networks and perform real-time generation of data products for downlink to information subscribers and application developers

- Demonstration and evaluation of NASA's next-generation CubeSat-sized, rad-tolerant, high-performance computer known as SpaceCube v3.0 Mini [2] including fault-tolerant computer architecture design and mitigation strategies, and several other CS2-compatible cards

Extended objectives for SCENIC include the ability to upload future HSI-based applications for additional selected science and defense applications trained with the downlinked dataset. Finally, SCENIC will also provide flight validation of new CubeSat form-factor guidance and navigation cards to be used on future NASA missions.

Planned concept-of-operations for SC-LEARN on SCENIC includes reconfiguring the device and retraining the TensorFlow Lite model. This process will involve capturing data products and building a training dataset from the HSI sensor, re-training the model on the ground, converting the model to a TensorFlow Lite model, uploading the TensorFlow Lite model to the onboard processor, and then reprogramming the Edge TPU.

Hyperspectral Models

In preparation for onboard operations on STP-H9/SCENIC, we examined two hyperspectral models from literature. There were two main selection criteria for the models: (1) their compatibility with the Edge TPU's set of supported operations and (2) their reported high accuracy on publicly available hyperspectral datasets.

The first model corresponds to the 1D multi-layer perceptron (MLP) implemented in [26]. In this paper, the authors performed a review of multiple state-of-the-art deep neural-network models. From their findings, the 1D MLP had the highest classification accuracy on the Indian Pines dataset, while a 3D convolutional neural network (CNN) had the highest classification accuracy on Pavia University. Unfortunately, the 3D CNN model is not compatible with the Edge TPU since 3D convolutions are not supported on the device. The implementation of the 1D MLP used in this research was adapted from their open-source repository, DeepHyperX[1]. To stabilize training, we inserted batch normalization layers after each fully connected layer, and to reduce overfitting of the model, weight decay was used as a regularization method. The 1D MLP model architecture diagram is shown in Figure 2.

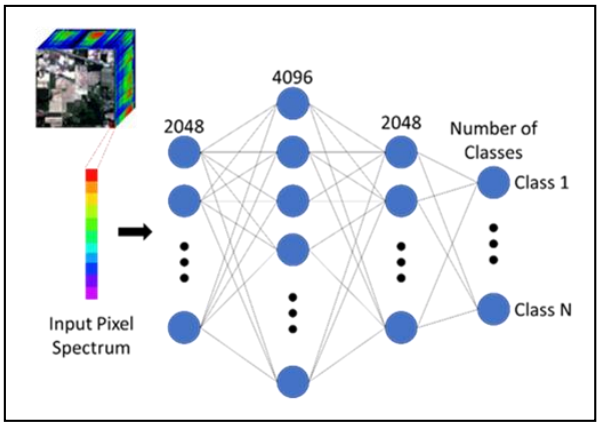

Figure 2: 1D Multi-Layer Perception Model

The second model we investigated was adapted from [28], a spectral–spatial CNN (SS-CNN) for hyperspectral image classification. In this model, features are extracted using two branches: (1) the first branch is a 1D CNN on the spectral information at a particular pixel, and (2) the second branch is a 2D CNN on the mean of the spatial patch surrounding the pixel. The features from these branches are then concatenated and input to a two-layer fully connected network, which outputs a prediction. Unlike [28], we included weight decay for regularization purposes. The SS-CNN model architecture diagram is shown in Figure 3.

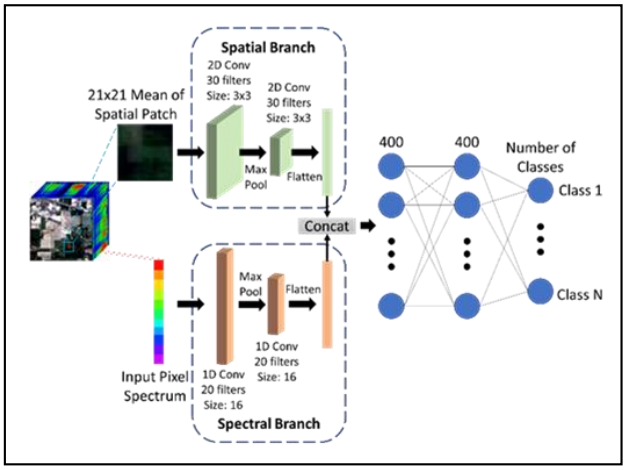

Figure 3: Spectral-Spatial Feature Learning Model

III. DESIGN

This section describes the design approach for creating the SC-LEARN and integrating it into the STP-H9 payload. Additionally, we describe the supporting test card design for prototyping experiments on the SC-LEARN and communicating with testbed hardware.

[1] https://github.com/nshaud/DeepHyperX

SC-LEARN Architecture Overview

The SC-LEARN is designed to act as a co-processor to expand the capabilities of NASA's existing high-performance space processors. The host processor configures, controls, reprograms, and feeds data to the Edge TPUs on the SC-LEARN to change different applications dynamically, such as switching from disaster detection mode (e.g., earthquakes, floods, and fires) to highly accurate targeting modes. More specifically, the SC-LEARN features three Edge TPU Accelerator Modules in a 1U CubeSat form-factor where each Accelerator Module is a multi-chip device that features the Edge TPU accelerator ASIC, power circuitry, and an internal reference clock. The Accelerator Modules are powered by three independent rad-hard load switches, which are controlled by the host processor. The load switches incorporate current sense amplifiers whose output is monitored by an onboard rad-hard analog-to-digital (ADC) converter. In the event there is a fault-induced current spike on one of the Accelerator Modules, the load switch will disable power to that module while the others can remain functional. In addition to integrated temperature sensing, the SC-LEARN incorporates external thermistors connected to the onboard ADC to monitor temperature. A high-level architecture block diagram of the SC-LEARN components is illustrated in Figure 4.

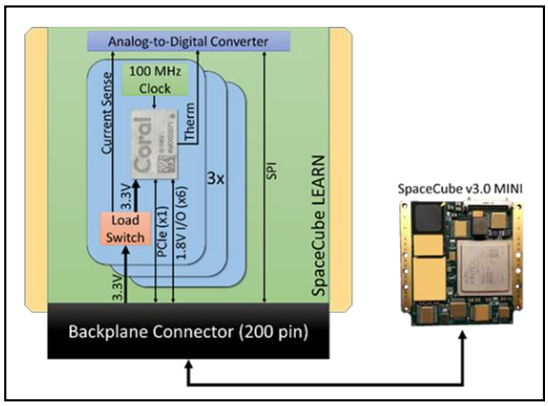

Figure 4: SC-Learn Architecture Block Diagram

The fabricated and assembled design is featured in Figure 5, where the SC-LEARN is inserted into another card that contains connectors to interface with the automated safe-to-mate (ASTM) system. The ASTM will check to ensure that there are no shorts, opens, or swapped connections in an assembled PCB or harness, thus verifying that the Device Under Test (DUT) is safe to integrate with the rest of the system, or to perform initial power-on testing.

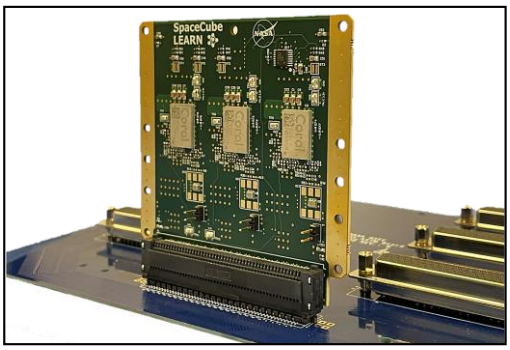

Figure 5: SC-LEARN plugged into Automated Safe-to-Mate (ASTM) Card

SPECULATE: Test and Evaluation Board

For ground-based testing, the SC-LEARN connects to an evaluation or adapter board with standard interfaces for rapid desktop prototyping. This adapter card is known as SPECULATE (SPacE CUbe LeArn TEst board), which is used to power the SC-LEARN and interface to multiple host FPGA processors, most notably the SpaceCube v3.0 Mini. SPECULATE can integrate with multiple FPGA development boards using the FPGA Mezzanine Card (FMC) connector, a common interface provided on many Xilinx development cards and on the active evaluation board for SpaceCube v3.0 Mini. The FMC connector is the main interface to the host FPGA board and includes PCIe, USB 2.0, and multiple IO control signals for the Edge TPUs on the SC-LEARN. The board can be powered through several options including the FMC card, a set of banana jacks, or a standard 12V "wall wart" connection. Figure 6 pictures a rendered model of the SC-LEARN inserted into the SPECULATE adapter.

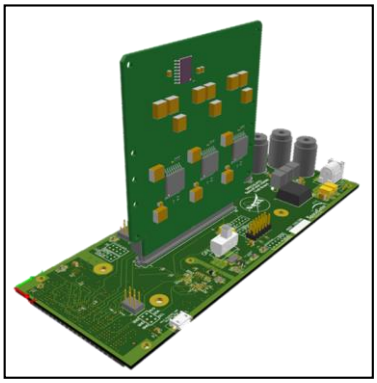

Figure 6: SC-LEARN plugged into SPECULATE FMC Evaluation Card

Table 2: Comparison of Model Complexity

Model	Dataset	# trainable parameters	On-Chip Memory for Caching Parameters	Off-Chip Memory for Streamed Parameters
MLP	Indian Pines	17,244,176	536.25KiB	16.02MiB
MLP	Salinas	17,244,176	536.25KiB	16.02MiB
MLP	Pavia U	17,031,177	344.25KiB	16.02MiB
SS-CNN	Indian Pines	948,106	3.27MiB	320.00B
SS-CNN	Salinas	948,106	3.27MiB	320.00B
SS-CNN	Pavia U	785,299	3.14MiB	320.00B

IV. OPERATIONAL MODES

SC-LEARN purposely includes three Edge TPU Accelerator Modules to enable several different operational modes in flight. These modes are: (1) a high-performance parallel-processing mode, (2) a fault-tolerant design mode, and (3) a power-saving mode with cold spares.

High-Performance Capability Mode

Multiple Accelerator Modules can be used cooperatively to execute operations in parallel to dramatically improve performance over a single-node design. Inference is especially amenable to simultaneous, parallel execution. To measure the expected performance of the parallel-processing SC-LEARN configuration, an application was developed to perform inference on a varying number of Edge TPUs for both previously described hyperspectral classification models. For this application, the trained models were first quantized and compiled for the Edge TPU. For more detail on the integer quantization process, we refer the reader to Section V. Table 2 compares the complexity of each of the trained models in terms of their number of parameters and how the Edge TPU caches the model parameters. The MLP model is considerably larger than the SS-CNN model with nearly 18× the number of parameters. Due to its relatively smaller size, the SS-CNN model parameters can be primarily stored in the on-chip memory of the Edge TPU with very small amounts of data required to be streamed in from the host's memory. In contrast, the MLP model primarily uses the host memory to stream in the model parameters because it exceeds the storage capacity of the Edge TPU's cache.

The application was executed on an increasing number of samples to examine how the execution time scaled with expanding amounts of data. In the case of a single Edge TPU Accelerator Module, inference was performed on the samples for a baseline comparison. For the multiple Edge TPU cases, threads were spawned corresponding to the number of available Edge TPUs and the samples were split evenly across them for inference.

Multilayer Perceptron Results: Figure 7 through Figure 9 show execution time for the MLP model scaled with the number of samples for different numbers of connected Edge TPUs for the Indian Pines (Figure 7), Pavia University (Figure 8), and Salinas datasets (Figure 9). Comparing the MLP execution time of one to two Edge TPUs across the three datasets, the use of two Edge TPUs is only slightly slower than one Edge TPU for a small number (<50) of samples. However, as the number of samples increases beyond 500, the speedup approaches near linear scaling, executing around twice as fast using two Edge TPUs compared to one. Therefore, the processing and communication overhead of spawning two threads appears to ameliorate as the number of samples increases. However, upon adding a third Edge TPU, a dramatic decrease in performance is observed. On further examination, we speculate that performance is highly limited by the Edge TPU runtime software and the communication overhead for the Edge TPU's driver when communication exceeds two Edge TPUs.

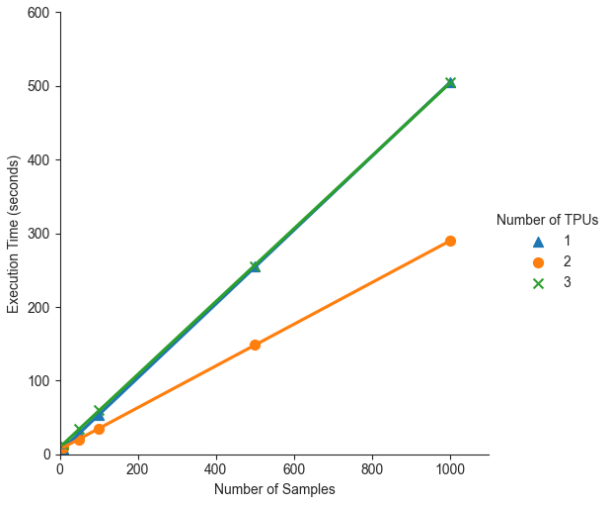

Figure 7: MLP Execution on Indian Pines Dataset

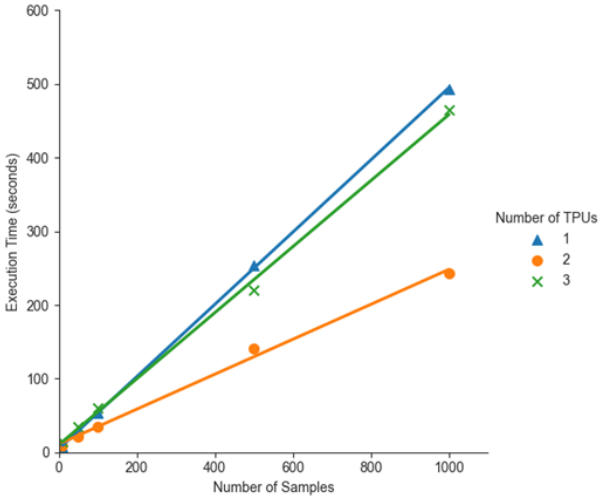

Figure 8: MLP Execution on Pavia University

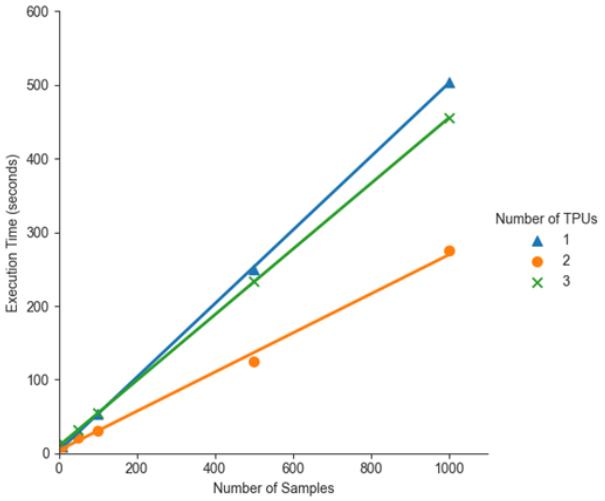

Figure 9: MLP Execution on Salinas Dataset

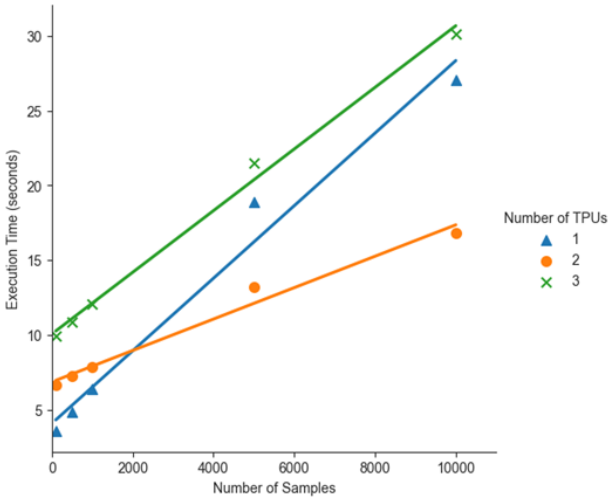

Figure 10: SS-CNN Execution on Indian Pines

Spectral-Spatial Results: Figure 10 through Figure 12 show the execution time for the SS-CNN model scaled with the number of samples for different numbers of connected Edge TPUs for the Indian Pines (Figure 10), Pavia University (Figure 11), and Salinas datasets (Figure 12). Comparing the SS-CNN execution time with the MLP, the SS-CNN runs much faster than the MLP, and therefore, its throughput is much higher. For example, in the case of 100 Salinas samples executed on a single Edge TPU, the SS-CNN executes 15× faster than the MLP. The reason for this vast difference in performance between the two models is, according to Table 2, that the SS-CNN parameters can fit fully in the Edge TPU's cache while the MLP parameters cannot due to its larger memory footprint. As a result, the Edge TPU does not cache the MLP parameters on-chip, but rather, streams them from the host's memory, causing the execution to be bottlenecked by the communication link between the host and Edge TPU.

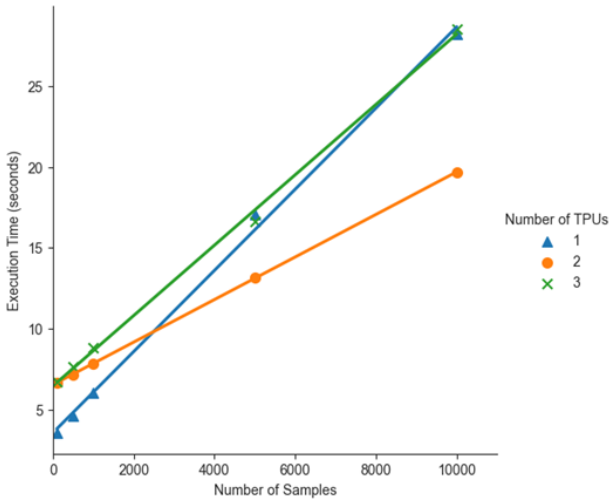

Figure 11: SS-CNN Execution on Pavia U Dataset

Unlike the MLP model, when comparing the SS-CNN execution times for one to two Edge TPUs across the three datasets, one Edge TPU is substantially faster than two Edge TPUs for a smaller number of samples (less than approximately 2000). For example, in the case of 100 Salinas samples, SS-CNN inference with one Edge TPU is 1.86× faster than SS-CNN inference with two Edge TPUs. This slow down for the two Edge-TPU case is likely caused by the overhead of spawning two threads and sending the model to two Edge TPUs. However, as the number of samples increases beyond approximately 2000, the benefit from distributing the data across two Edge TPUs outweighs the processing and communication overhead associated with spawning two threads. Similar to the MLP models, upon adding a third Edge TPU, the performance dramatically decreases.

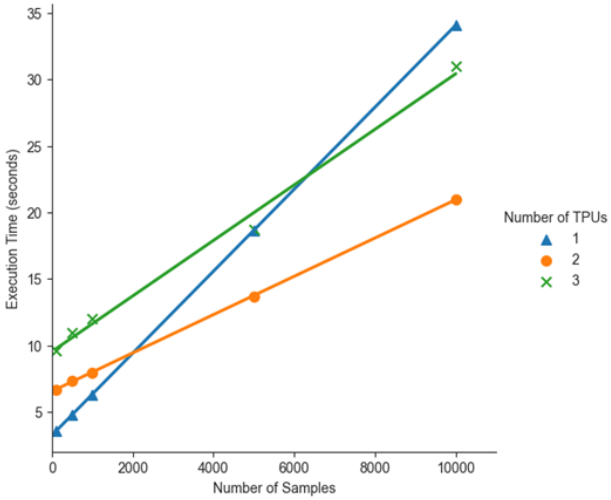

Figure 12: SS-CNN Execution on Salinas Dataset

Fault-Tolerant Design Mode

One of the most common fault-tolerant techniques is employing hardware redundancy to mitigate failures. The most prominent of these hardware techniques is known as Triple-Modular-Redundancy (TMR), a mode in which the output of three replicas of a device are run through a majority voter for fault masking. In this configuration, two of the devices must fail for an error to propagate. This design mitigation was desirable since preliminary analysis of the SC-LEARN in mission scenarios demonstrate it would be unlikely for two devices to be simultaneously affected by radiation-induced single-event upsets. The SC-LEARN operates in tandem with a host processor, therefore it is also essential that the host design, acting as the majority voter, also has redundancy. Both the SpaceCube v3.0 Mini and SpaceCube Mini-Z/Z+ processor cards incorporate an interface to SC-LEARN through the FPGA design. The FPGA resources can be replicated using hardware-redundancy methods described in [27], and all the FPGA designs additionally include various configuration-memory scrubbing techniques for repairing single-event upsets caused by the radiation environment. For verification of this mode, we developed a host application that spawned three threads corresponding to the three connected Edge TPUs. Each thread transmitted the same data to its corresponding Edge TPU for inference. Upon completion of inference, each Edge TPU returned its classification result to the host, which then compared the results from all three.

In addition to hardware redundancy, the reliability of the SC-LEARN design is reinforced with quality part selection and independent monitoring. Each of the three Edge TPUs include individual load switches controlled and monitored by the host processor. Therefore, the host processor can intervene with corrective measures if a high-current event is detected. Finally, the SC-LEARN design incorporates high-reliability power distribution components and space-grade passives to reduce possible sources of failure.

Power-Saving Mode

To conserve onboard power and provide the final operational mode, the SC-LEARN design benefits from individual load switches for each Edge TPU accelerator. The card can operate in a lower power mode state when two of the three Edge TPUs are depowered allowing for cold sparing of the system. In this mode, the system host has the ability to enable any one of the three redundant Edge TPU Accelerator Modules. This allows mission operators to select an alternative accelerator should one become damaged over the lifetime of the mission.

V. TRAINING AND QUANTIZATION

In this section, we address the model-training process for each of the test datasets. Additionally, we review the quantization process and note the impacts and considerations for employing this technique for use in space.

Training Process

As described in Section II, both the 1D MLP and SS-CNN models were trained on the Salinas, Indian Pines, and Pavia University datasets. For this training, the Adam optimizer [29] was used in TensorFlow/Keras. For each dataset, 76% of the labeled data was randomly selected to establish the training set, 4% for the validation set, and finally, 20% for the test set. The number of training epochs was fixed to 50 for the training and validation loss to sufficiently converge. The models were trained with five different learning rates: 0.1, 0.01, 0.001, 0.0001, and 0.00001. Among these five models trained with different learning rates, the one producing the highest accuracy on the test set was selected.

Quantization Process

Edge and embedded devices frequently have limited resources, especially for memory capacity and computational power. Developers often employ varying optimization strategies to TensorFlow models to reduce the burden on the available onboard resources. One of the most frequent, and in the case of the Edge TPU Accelerator Module, required strategies is model quantization. Quantization is useful as a general technique for AI models because it can reduce inference latency, power, and model size with relatively low degradation in model accuracy.

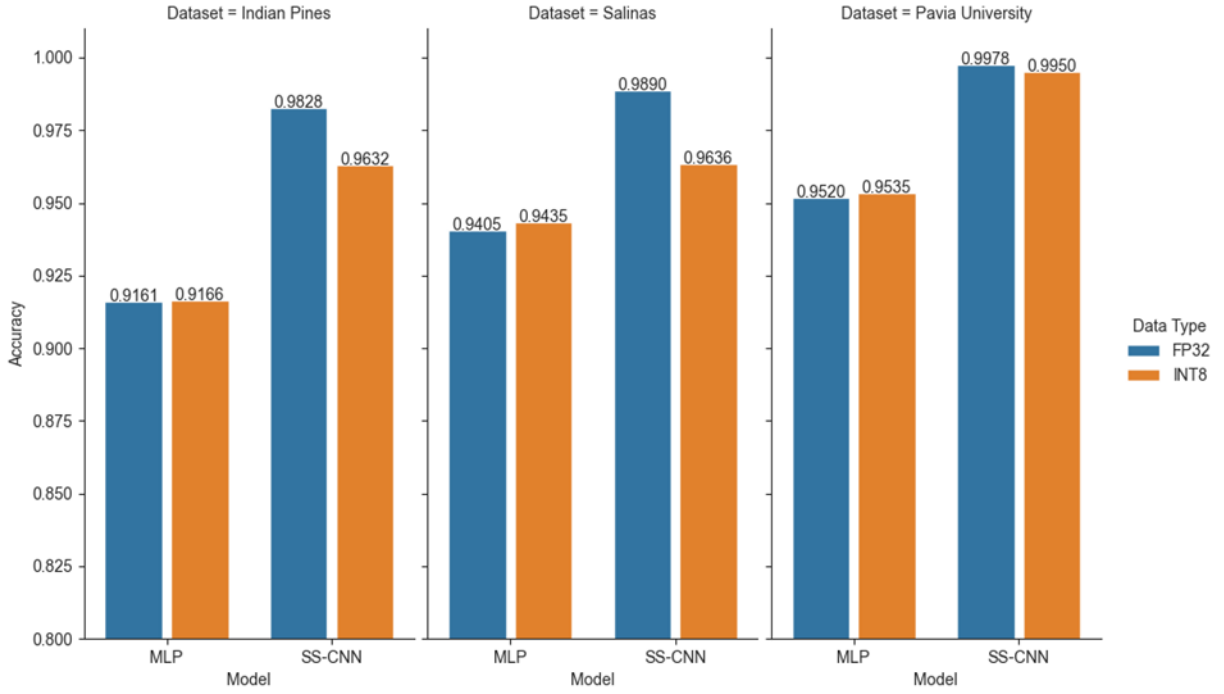

Figure 13: Comparison of Quantized and FP Accuracies in Evaluated Datasets

While the training process used 32-bit floating-point (FP32) data types for all weights and tensors, the Edge TPU can only operate on 8-bit integer (INT8) weights and tensors—an intended architectural design decision to decrease inference latency and power consumption. As such, post-training 8-bit quantization was performed using the TensorFlow Lite converter [31] to convert FP32 operations to INT8 operations. Consequently, reducing the precision of the weights and tensors can lead to a decrease in accuracy of the model. However, provided a representative dataset, post-training quantization has been shown to minimally decrease accuracy while providing substantial decreases in inference latency [30]-[31]. The TensorFlow Lite converter specification uses Equation 2 to approximate floating-point values:

$$Value_{FP32} = (Value_{INT8} - z_0) * s \quad (2)$$

where z_0 is the zero point (INT8) and s is a scale factor (FP32). Activations are asymmetric (i.e., their zero point is non-zero) while the weights in the TensorFlow Lite specification are forced to be symmetric (i.e., their zero point is equal to 0). To optimally select the zero point and scales to limit accuracy degradations of the quantized model, a representative dataset must be given to the TensorFlow Lite converter to estimate the dynamic range of the activations. For the hyperspectral models, we provided the full training dataset (76% of the data) as the representative dataset.

After performing quantization, we measured the accuracy of FP32 and INT8 models on the test set for each dataset. The results, displayed in Figure 13, show the SS-CNN model outperforms the MLP model for both FP32- and INT8-quantized models over all three datasets in terms of accuracy. The outcome indicates that incorporating spatial information aids in classifying hyperspectral data. Comparing the FP32 models to their respective INT8-quantized counterparts, the INT8-quantized SS-CNN incurred 2.54% and 1.95% decreases in accuracy on the Salinas and Indian Pines datasets, respectively. However, for the Pavia University dataset, there is only a slight accuracy loss of 0.28%. In contrast, the MLP model does not suffer accuracy loss from post-training integer quantization. Instead, the classification accuracy increases very slightly for INT8-quantized MLP models. The difference in accuracy losses due to integer quantization between the two models is most likely due to the differences in their model size (e.g. number of parameters) and architecture. Since the MLP model only uses dense layers, it has more redundant connections than the SS-CNN model, which uses a combination of convolutional layers and dense layers with much fewer neurons than the MLP model's dense layers. As a result of the larger number of redundant

connections in the MLP model's architecture, the MLP model's sensitivity to precision losses in weights and activations is lesser than the SS-CNN model.

The demonstration comparing both FP32 and INT8 is significant because for STP-H9, the Intel Myriad X can operate with FP32, while both the Edge TPU Accelerator Modules and the Xilinx DPU rely on quantized INT8 models. Detailed reliability studies have not been conducted to differentiate which AI architecture will be more reliable under specific environmental conditions. However, the results demonstrate that differences in model accuracy between the devices will not likely be a leading discriminator in future evaluations.

Future Work

In the future, to limit the accuracy degradation of the quantized models, we will investigate quantization-aware training [30]. In this process, quantization nodes are inserted into the training graph of the model to simulate the noise effects of quantization. As a result, the model can be optimized to be resilient to the effects of quantization during the training process.

VI. RADIATION TESTING

Preliminary reports for radiation testing with the Edge TPU have been shared with NASA; however, they cannot be readily disclosed. Therefore, NASA is independently pursuing testing and publication of the Edge TPU Accelerator Modules as part of the NASA Electronic Parts and Packaging Program (NEPP) with the NASA Radiation Effects Analysis Group (REAG). Both upcoming cumulative radiation damage (total ionizing dose) and heavy-ion single-event effect testing are planned depending on facility availability and will be included with regular NEPP updates.

Future Work

In the future, we will additionally investigate the use of fault-aware training (FAT) to mitigate the radiation effects of the space environment. This methodology, presented in [33], demonstrates that highly accurate neural networks can be trained to exhibit higher error tolerance compared to the original model without FAT.

VII. CONCLUSION

This research can provide the foundational platform for enabling onboard AI applications in a 1U CubeSat form-factor design with the SC-LEARN card. This development integrates a commonly used Google Coral Edge TPU AI platform that scientists and software developers can immediately purchase and begin prototype development on, without concerns about the absence of path-to-flight options. The SC-LEARN provides a reliable architecture and multiple operational modes for inclusion in future advanced NASA missions.

This paper describes both the SC-LEARN architecture design and the supporting ground-support equipment FMC evaluation card, known as SPECULATE, for immediate rapid prototyping of machine-learning applications. Additionally, this paper highlights two models, 1D MLP and SS-CNN, to be deployed for HSI experiments onboard the ISS as part of the STP-H9/SCENIC experiment. Finally, this research showed preliminary encouraging conclusions for devices relying on quantization of deep-learning models on an HSI dataset. The Edge TPU Accelerator Module is a readily available and widely usable design, which enables exploration of deep-learning models to be included in new mission and instrument analysis proposals.

Planned future work for the SC-LEARN includes both more readily available designs for testing with several development boards and examination of more detailed power use cases. The SC-LEARN and SPECULATE card can be integrated into many development platforms due to the adaptability of the FMC connector. Therefore, in addition to the reference designs for the SpaceCube v3.0 Mini and Xilinx KCU105 development board, future designs will also target the Mini-Z/Z+ along with their development boards (e.g.,Xilinx ZC706 and Diligent Zedboard / MicroZed Boards). Finally, additional applications will be developed to exercise a broad spectrum of power use cases to better characterize the power efficiency of the SC-LEARN design.

Acknowledgments

The authors would like to recognize the contributions and support by additional team members of the Science Data Processing Branch Code 587 and Goddard collaborators including Nicholas Franconi, Alessandro Geist, and Michael Lin. The authors also recognize assistance from Kristy Sakano from NAVAIR. The authors thank our NASA Goddard Planetary Science collaborators for support including Nicolas Gorius, Shahid Aslam, Tilak Hewagama, and Thanh Nguyen. We also thank our supporting partners at the NASA Electronics Parts and Packaging Program, Ed Wyrwas, Megan Casey, and Jonny Pellish. For accommodation for the SCENIC experiment, we acknowledge the great team at STP-Houston and AFRL Collaborators Tyler Lovelly, Josh Donckels, and Jesse Mee. Finally, special thanks to our key sponsor supporting this development, the NASA/GSFC Internal Research and Development (IRAD) program.

References

1. "Google Coral," Google. [Online]. Available: https://coral.ai/products/
2. Brewer, C., Franconi, N., Ripley, R., Geist, A., Wise, T., Sabogal, S., Crum, G., Heyward, S., and C. Wilson, "NASA SpaceCube Intelligent Multi-Purpose System for Enabling Remote Sensing, Communication, and Navigation in Mission Architectures," 34th Annual AIAA/USU Conference on Small Satellites, SSC20-VI-07, Logan, UT, Aug. 1-6, 2020.
3. Ripley, R., Fraction, J., Soto, L. S., Clagett, C, Brewer, C., and A. Geist, "Modular Architecture for a Resilient Extensible SmallSat (MARES)" 34th Annual AIAA/USU Conference on Small Satellites, Poster 207, Logan, UT, Aug. 1-6, 2020.
4. "NASA Strategic Technology Investment Plan," NASA Office of the Chief Technologist, August 2017.
5. "2015 NASA Technology Roadmaps," NASA Office of the Chief Technologist, July 2015.
6. "2020 NASA Technology Taxonomy," Washington, D.C., USA: NASA Office of the Chief Technologist, 2020.
7. National Academies of Sciences, Engineering, and Medicine, "Thriving on Our Changing Planet A Decadal Strategy for Earth Observation from Space," Washington, DC: The National Academies Press. 2018. https://doi.org/10.17226/24938
8. National Academies of Sciences, Engineering, and Medicine, "Visions into Voyages for Planetary Science in the Decade 2013-2022: A Midterm Review," Washington, DC: The National Academies Press. 2018. https://doi.org/10.17226/25186
9. Edwards, C., and et al. "Emerging Capabilities for Mars Exploration," Planetary Science Decadal, White Paper, 2020.
10. Cardillo, R., "Small Satellite 2017 Keynote Address," 31st Annual AIAA/USU Conference on Small Satellites, Logan, UT, Aug 7, 2017. https://www.nga.mil/MediaRoom/SpeechesRemarks/Pages/Small-Satellites---Big-Data.aspx
11. Air Force Space Command, "AFSPC Long-Term Science and Technology Challenges," Colorado Springs, CO: Peterson AFB, 2016.
12. DARPA, "Blackjack Pit Boss," Arlington, VA: DARPA Tactical Technology Office, Broad Agency Announcement, HR001119S0012, April 2019.
13. "Final Report," National Security Commission on Artificial Intelligence, 2021. [Online]. Available: https://reports.nscai.gov/final-report/table-of-contents/
14. Chien, S., et al. "The EO-1 Autonomous Science Agent", 3rd International Joint Conference on Autonomous Agents and Multi-Agent Systems, New York, NY, July 18-23, 2004.
15. Thompson, D. R., Altinok, A., Borstein, B., Chien, S. A., Doubleday, J., Bellardo, J., and K. L. Wagstaff, "Onboard Machine Learning Classification of Images by a Cubesat in Earth Orbit," AI Matters, vol. 1, no. 4, pp 38-40, Jun. 2015.
16. Abbey, J., and S. Boland, "On-orbit Calibration of Magnetometer Using Stochastic Gradient Descent," 33rd Annual AIAA/USU Conf. on Small Satellites, SSC19-WKII-02, Logan, UT, August 3-8, 2019.
17. Ghassemi S., and E. Magli, "Convolutional Neural Networks for On-Board Cloud Screening," MDPI Journal of Remote Sensing, vol. 11, no. 1417, pp. 1-14, Jun. 2019.
18. Lofqvist, M., and J. Cano, "Accelerating Deep Learning Applications in Space," 34th Annual AIAA/USU Conf. on Small Satellites, SSC20-WKIV-01, Logan, UT, August 1-6, 2020.
19. Rughani, R., and D. Barnhart, "Safe Construction in Space: Using Swarms of Small Satellites for In-Space Manufacturing," 34th Annual AIAA/USU Conf. on Small Satellites, SSC20-WKVI-01, Logan, UT, August 1-6, 2020.
20. Giuffrida, G., Diana, L., de Gioia, F., Benelli, G., Meoni, G., Donati, M., and L. Fanucci, "CloudScout: A Deep Neural Network for On-Board Cloud Detection on Hyperspectral Images," MDPI Journal of Remote Sensing, vol. 12, no. 2205, May 2020.
21. Esposito, M., Conticello, S. S., Pastena, M., and B. Carnicero Domínguez, "In-orbit demonstration of artificial intelligence applied to hyperspectral and thermal sensing from space," Proc. SPIE 11131, CubeSats and SmallSats for Remote Sensing III, 111310C (30 August 2019); doi: 10.1117/12.2532262 https://doi.org/10.3390/rs12142205
22. "Computer solutions for SpaceCloud," Unibap, [Online]. Available: https://unibap.com/en/our-offer/space/spacecloud-solutions/

23. Grana, M, Veganzons, MA., and B. Ayerdi, "Hyperspectral Remote Sensing Scenes," Grupo de Inteligencia Computacional (GIC), [Online]. Available: http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes

24. Sims, E., "The Department of Defense Space Test Program: Come Fly with Us," Proceedings of IEEE Aerospace Conference, March 2009.

25. Xilinx, "Zynq DPU v3.3," Xilinx Corporation, PG338 v3.3, Feb. 3, 2021.

26. Audebert, N., Le Saux, B.., and S. Lefevre, "Deep Learning for Classification of Hyperspectral Data: A Comparative Review," in IEEE Geoscience and Remote Sensing Magazine, vol. 7, no. 2, pp. 159-173, June 2019.

27. Wirthlin, M, "High-Reliability FPGA-Based Systems: Space, High-Energy Physics, and Beyond," Proceedings of the IEEE, vol. 103, no. 3, pp. 379-389, March 2015, doi: 10.1109/JPROC.2015.2404212

28. Deng, C., Xue, Y., Liu, X., Li, C., and T. Dacheng, "Active Transfer Learning Network: A Unified Deep Joint Spectral–Spatial Feature Learning Model for Hyperspectral Image Classification," IEEE Transactions on Geoscience and Remote Sensing, vol. 57, no. 3, Mar. 2019.

29. Kingma, D. and J. Ba, "Adam: A Method for Stochastic Optimization," 3rd International Conference for Learning Representations, San Diego, Ca, 2015.

30. Jacob, B., Kligys, S., Chen, B., Zhu, M., Tang, M., Howard, A., Adam, H., and D. Kalenichenko, "Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference" arXiv preprint arXiv: 1712.05877 [cs.LG], 2017.

31. *Post-training quantization*, TensorFlow, April 7, 2021. [Online]. Available: https://www.tensorflow.org/lite/performance/post_training_quantization

32. Hashemi, S., Antony, N., Tann, H., Bahar, R. I., and S. Reda, "Understanding the Impact of Precision Quantization on the Accuracy and Energy of Neural Networks" Proceedings of the Conference on Design, Automation & Test, pp. 1478–1483, Mar. 2017.

33. Zahid, U., Gambardella, G., Fraser, N. J., Blott, M., and K. Vissers, "FAT: Training Neural Networks for Reliable Inference Under Hardware Faults," arXiv preprint arXiv:2011.05873v1 [cs.LG], 2020.