High Energy and Thermal Neutrons Sensitivity of Google Tensor Processing Units

High Energy and Thermal Neutrons Sensitivity of Google Tensor Processing Units

Rubens Luiz Rech Junior*, Sujit Malde†, Carlo Cazzaniga†, Maria Kastriotou†,
Manon Letiche‡, Christopher Frost†, and Paolo Rech*§

***Abstract*—In this paper we investigate the reliability of Google's Coral Tensor Processing Units (TPUs) to both high energy atmospheric neutrons (at ChipIR) and thermal neutrons from a pulsed source (at EMMA) and from a reactor (at TENIS). We report data obtained with an overall fluence of $3.41 \times 10^{12} n/cm^2$ for atmospheric neutrons (equivalent to more than 30 million years of natural irradiation) and of $7.55 \times 10^{12} n/cm^2$ for thermal neutrons. We evaluate the behavior of TPUs executing elementary operations with increasing input sizes (standard convolutions or depthwise convolutions) as well as eight CNNs configurations (SSD MobileNet v2 and SSD MobileDet, trained with COCO dataset, and Inception v4 and ResNet-50, with ILSVRC2012 dataset). We found that, despite the high error rate, most neutrons-induced errors only slightly modify the convolution output and do not change the CNNs detection or classification. By reporting details about the error model we provide valuable information on how to design the CNNs to avoid neutron-induced events to lead to miss detections or classifications.**

I. Introduction

Convolutional Neural Networks (CNNs) are today the most effective (and efficient) way to detect an object in a scene. By applying various filters to the input image, convolutional layers extract information (feature maps) that is then passed to the downstream layers to detect and/or classify objects. The number of layers, the kind of filter applied, and the structure of the CNN is engineered to achieve the desired accuracy and efficiency. The prediction process is highly computational demanding, as it is necessary to apply several filters to each feature map. The filtering process is mapped into a matrix multiplication operation, which can be efficiently executed in parallel accelerators, such as Graphics Processing Units (GPUs) or Field Programmable Gate Arrays (FPGAs).

To ensure very high accuracy along with real-time detection (at least 40 frames per seconds must be processed), both being fundamental for autonomous vehicles, it is necessary to execute CNNs on highly performant, costly, and power hungry devices, such as the latest GPUs or very big FPGAs. Nevertheless, the field of adoption of CNNs is not limited to self-driving cars. Many other applications, with less strict accuracy and timing constraints, can benefit from CNNs execution. This is the case of Internet of Things (IoT), smart homes, or smart cities, in which detecting or identifying a relatively low number of objects can significantly improve the user experience and the overall system features. In these applications the cost and power consumption must be minimized, while still guaranteeing sufficient accuracy.

Lately, vendors have developed low-cost accelerators for CNNs execution, named *EdgeAI* devices, such as NeuroShield or Google Coral Tensor Processing Units (TPU). These EdgeAI devices are only able to execute elementary operations (i.e., convolutions and some other matrices operations) in low precision (16-bit floating point or even 8-bit integer). Coupled with a good software framework (e.g., Tensor Flow) that runs on a host device, EdgeAI devices significantly reduce the time and power consumption of the convolution, which is the most computational demanding operation of CNNs. As EdgeAI devices are likely to be used at scale and in distributed systems, it is fundamental to investigate their reliability, in particular their neutron-induced error rate. Preliminary studies showed that, despite being small, EdgeAI devices have a not negligible neutrons- or protons-induced error rate [1], [2].

In this paper, we investigate the reliability to neutrons of Google Coral TPU. Unlike previous works on EdgeAI reliability, we deeply investigate the fault model on the main elementary operations (standard and depthwise convolutions). Moreover, we compare the error rate and the prediction failures of eight CNNs configurations: SSD MobileNet v2 and SSD MobileDet, trained with COCO dataset, as well as Inception v4 and ResNet-50, trained with ILSVRC2012 dataset.

To have a broad evaluation, we test the Coral TPU with both high energy neutrons, at the ChipIR facility, and with thermal neutrons, at the EMMA facility in UK and at TENIS facility at Institut Laue-Langevin (ILL) in Grenoble, France. While the high-energy neutrons cross section of the Coral TPU is much higher than the thermal neutrons cross section, regardless of the type of neutrons, the results are consistent in the sense that depthwise convolutions are shown to have higher error rate than standard convolutions and SSD MobileDet is less reliable than SSD MobileNet V2.

The rest of the paper is organized as follows. In Section II, we provide a solid background on CNNs and on the hardware and software architecture of the Coral TPU, useful to under-

Rubens Luiz Rech Junior and Paolo Rech are with the Instituto de Informatica, Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, Brazil, email: {rlrech, ffsantos, prech}@inf.ufrgs.br; Paolo Rech is also with DAUIN, Politecnico di Torino and with DII, Università di Trento; Sujit Malde, Carlo Cazzaniga, Maria Kastriotou, and Christopher Frost are with ISIS Facility, UKRI-STFC, United Kingdom; Manon Letiche is with TENIS, Institut Laue-Langevin, France. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 886202, from The Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - Finance Code 001, and from the French national program Programme d'Investissements d'Avenir, IRT Nanoelec ANR-10-AIRT-05. Neutron beam time was provided by ChipIR (DOI: 10.5286/ISIS.E.RB2000161) and ILL (TEST-3203).

stand the experimentally observed behaviors. In Section III, we describe the high energy and thermal neutrons setups we developed and the software (convolutions and CNNs) we test. Experimental results are presented and discussed in Section IV, highlighting the implications for future hardening solutions for Coral TPU, while Section V concludes the paper.

II. Background

In this Section, we review the main characteristics of CNNs, the architecture of EdgeAI devices (focusing on the Coral TPU), and the software framework used to train and execute CNNs on EdgeAI accelerators.

A. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are today widely adopted to perform object detection [3]. One of the key steps when using CNNs for object detection is *convolution*. A CNN is a sequence of layers of different kind, each applying a specific function to the input frame (or feature map).

Convolution layers are the computing core of CNN. By applying filters, convolution layers extract information from the input frame that is then processed to identify objects. More than 80% of the computation in a CNN is dedicated to convolution, which is why most device architects are focusing on making convolution more and more efficient, producing novel devices such as the Coral TPU.

Lately, it has been shown that the efficiency of CNNs execution can be significantly improved approximating operations [4] or hardware component [5], [6] and it has been shown that the same object detection accuracy can be achieved, through re-training, representing data in 16-bit floating-point [7], 8-bit integer, or even in binary values [8]. Most low-power accelerators take advantage of reduced-precision operations to reduce the computing power required to run CNNs. The Coral TPU we used in this study, for instance, executes operations in 8-bit integer.

B. EdgeAI accelerators

EdgeAI accelerators, like NeuroShield and Google Coral TPU, are low-power and low-cost devices designed to perform heavy machine learning computations in the context of embedded applications.

Figure 1 shows the high level schematic of the Coral TPU architecture which is mainly composed by a systolic array fed by a large set of input buffers (not protected by ECC). The array outputs the product of the model weights and each layer's input into the activation unit, where the partial sums are accumulated and the activation function is applied. Therefore, this device can perform a set of operations, mainly convolutions, which are a fundamental block for machine learning applications, in an extremely power- and performance-efficient manner, i.e., the TPU delivers 2 TOPS per watt.

For minimizing data transfers and storage and speed up calculations, all data that is computed and stored within the TPU is represented as 8-bit unsigned integers (UINT8). The

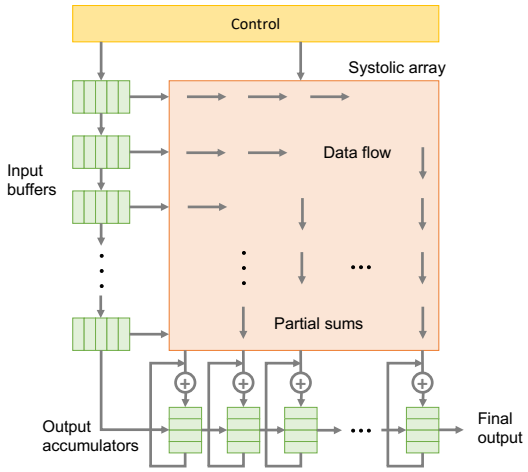

Fig. 1. High level schematic of the Coral Edge TPU architecture. Adapted from [9].

device is capable of performing the quantization and de-quantization steps for interfacing with the host floating-point representations.

Since Coral TPU is simply an accelerator, it must be connected to a host device. Google provides two versions of the accelerator: one that interfaces with the host via PCIe and the other uses USB 3.0. On our setup, we have a Raspberry Pi 4 as host, connected to the Coral USB accelerator.

The software layer of the Coral TPU is based on TensorFlow Lite, which is a light version, optimized for embedded devices, of the TensorFlow framework developed by Google for machine learning [10]. Most of the development effort is very similar as if the ML model would run on a normal CPU, however there is an EdgeTPU compiler that is responsible for deploying the TensorFlow Lite model targeting the Coral Edge TPU architecture.

C. CNNs Reliability

CNNs have already been shown to be particularly susceptible to transient faults [11], [12]. Through beam experiments and fault-injection, it has been demonstrated that the corruption of each layer has a different probability of affecting the CNN output, being the convolution layer the responsible for most observed errors [11]. The corruption of a layer or an operation inside a layer can be masked without affecting the output, can reach the output but keep the classification/detection unaltered, or can spread and modify the output in a way that impact the CNN functionality. Thanks to the intrinsic approximate nature of CNN computation, most of the errors do not turn into system failures, i.e., they do not affect the CNN accuracy. This has been proved for GPUs [11], FPGAs [13], and NeuroShield devices [1], [2]. Unfortunately, despite the intrinsic approximate nature, the misdetections and misclassifications rates in CNN executed in modern computing devices are still too high to be employed in safety-critical applications [11], [12]. As discussed in Section III-A, we distinguish between critical and tolerable errors in CNN execution on the Coral TPU. Additionally, we investigate the corrupted element distribution at the output of atomic convolutions.

The Coral TPU, to improve performances, executes operations in 8-bit unsigned integers. It has been shown that reducing operation precision, while bringing unquestionable benefits to efficiency, has the drawback of increasing the (negative) impact of a fault in the operation output [14]. For CNNs, precision reduction turns into a higher probability for a fault to modify detection. It has been shown that a fault in a FP16 CNN has ∼2x the probability of causing misdetection than a fault in a FP32 CNN [15]. Part of our contribution is to evaluate whether the execution of CNNs using 8-bit integer is harmful for the system reliability.

Recently, some works have discussed the reliability of EdgeAI devices to neutrons and protons, focusing specifically the Arduino NeuroShield [1], [2]. To the best of our knowledge, this is the first work presenting experimental data on Coral TPU devices error rate. Previous studies showed that the error rate of the small EdgeAI accelerators is far from being negligible (higher than 10^2 Failure In Time - FIT rates). Unlike previous publications, we engineered a setup to test also atomic operations performed by the accelerator (convolutions) with different sizes and depths (2D and 3D). This information is useful to deeply investigate the fault model induced by neutrons.

III. Methodology

In this Section, we detail the Coral TPU device we test, the software we run, and the (high energy and thermal) neutrons beam experiment setups.

A. Hardware and Software

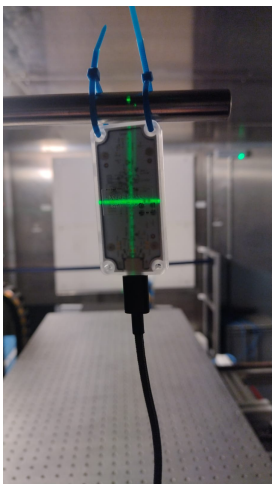

Fig. 2. The Coral TPU aligned with the high energy neutron beam at ChipIR. The host device, a Raspberry Pi 4, is connected with a 2 meters USB-3 cable and placed well out of the beam.

Coral Edge TPU is designed to accelerate machine learning algorithms, especially neural networks. Considering that most of the deep neural networks are fundamentally convolution operations, the atomic, basic operation of a Coral TPU is indeed convolution. As a first experiment, we want to evaluate the reliability of the two types of convolution that are supported by Coral: standard and depthwise. Standard convolutions are normal 2D convolutions while depthwise convolutions have an input composed by multiple channels and each one is convolved with its respective kernel separately. Since CNNs usually perform image prediction, in our experiments, the inputs of depthwise convolutions are always composed of three channels, as for the RGB colors, and this type is referred as 3D convolutions. We run tests with squared matrixes of sizes ranging from 256 to 1,250 (INT8) as inputs and squared kernels of fixed sizes: 40 for standard convolutions and 20 for the depthwise ones.

Besides convolutions, we evaluate the reliability of eight neural network configurations in which we vary the network architecture, the dataset, and the training methodology. We consider neural networks that perform the two main machine learning tasks supported by Coral: image classification and object detection. In image classification, the goal is to classify the object in the image, e.g., a dog, a car or a tree, without indicating the position. Object detection is a more complex task, as multiple objects in the image have to be located and then classified.

We consider four different network architectures. Two of them, Inception V4 [16] and ResNet-50 [17], target image classification. Both are trained with ILSVRC [18] dataset and support a wide range of 1,000 different object classes. The other two, SSDLite MobileDet [19] and SSD MobileNet V2 [20], perform object detection and are trained with COCO [21] dataset which embraces 90 classes. The models for these NNs are based on TensorFlow Lite.

In addition to these four models/configurations, we also retrain SSD MobileNet V2 with two other datasets: a subset of the COCO dataset (14 classes) and a subset of the Oxford-IIIT Pet [22] dataset (2 classes). Our goal is to evaluate whether and how a reduced number of objects to be detected impact the device error rate. The retraining process is done with and without the application of *transfer learning* technique, in which the knowledge from another machine learning model is reused in order to speed up the learning process.

Considering both types and multiple sizes of convolutions, as well as the different NN configurations, we provide experimental data obtained on 16 benchmarks.

B. Neutron Experiment Setup

Atmospheric neutrons experiments were performed at the ChipIR facility at the ISIS spallation neutron source of the Rutherford Appleton Laboratory (RAL), UK. ChipIR [23] is the reference beamline dedicated to the irradiation of microelectronics and it features a high energy neutron spectrum, as similar as possible to the atmospheric one. The flux with neutron energy above 10 MeV is $5.4\times10^6 n/cm^2/s$, while the thermal component ($E < 0.5eV$) is $4\times10^5 n/cm^2/s$ [24]. The Coral TPU was positioned 0.8m away from the ChipIR beamstop with a collimated beam size of 70×70 mm. A picture of the Coral TPU at ChipIR is shown in Figure 2. At the position of the Coral, the average flux was of about $3.9\times10^6 n/cm^2/s$. We test the device for more than 241 effective hours (without considering the setup, load input, download output, and reboot time), resulting in a fluence of more than $3.41 \times 10^{12} n/cm^2$.

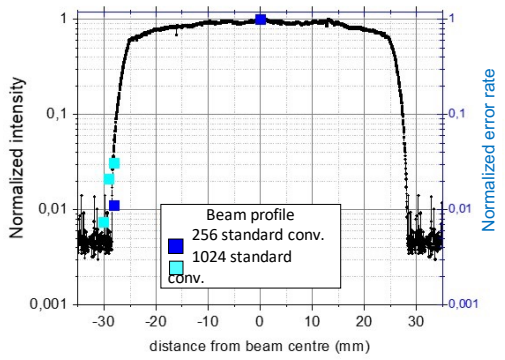

Fig. 3. Cross section of the beam profile at TENIS and the error rate for 256 and 1024 standard convolutions normalized to the error rate measured at the center of the beam. The beam is very stable in an area of 2cm x 2cm for then decreasing rapidly. If the flux at the center is too high the device can be placed at the beam edge to reduce the error rate.

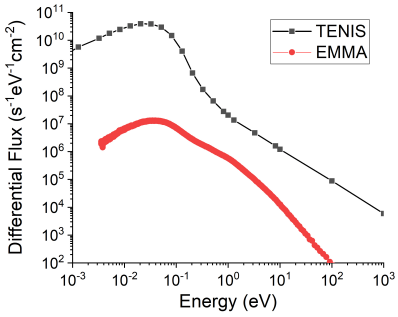

Fig. 4. Comparison of the neutron energy spectra of EMMA and TENIS. TENIS has a much higher component of epithermal neutrons.

When scaled to the terrestrial flux ($13n/cm^2/h$ [25]), this fluence correspond to more than 30 million years of natural irradiation.

The ISIS neutron source also feature various thermal neutrons facilities, such as the equipment materials and mechanics analyzer (EMMA) [26], that has a line of sight on the water moderator of the main neutron source. The thermal neutron beam is achieved from the pulsed neutrons source thanks to a *chopper* (a rotating device used to block a portion of the neutron beam in time) that is synchronous with the proton pulse, thus cutting the fast neutron portion of the spectrum, letting through only the thermal component. The thermal neutron flux delivered at EMMA is of about $2.32 \times 10^6 n/cm^2/s$. More details about the neutrons spectrum and the flux measurements at EMMA can be found in [26]. The availability of both high energy (ChipIR) and thermal (EMMA) neutrons facilities at ISIS is very convenient, as the same setup and the same devices can be tested back-to-back in both beam lines, allowing a direct comparisons of the sensitivity of the same device to two different neutrons spectra. Nevertheless, considering that thermal neutrons cross section is normally significantly lower than the high-energy-neutrons one, EMMA flux might be too low to test small configurations. We have used EMMA to characterize the TPU configurations with the higher error rate (MobDet and MobNet CNNs). After more than 25h of test at EMMA the 1024 convolutions provide only 10s of SDCs, making the characterization impractical.

To measure the thermal neutrons cross section of the TPU executing convolutions, that would not be possible at EMMA due to the low error rate, we also perform experiments at the new Thermal and Epithermal Neutron Irradiation Station (TENIS) hosted by the Institut Laue-Langevin (ILL). This new facility aims to replace D50 as a facility where thermal neutron experiments were conducted at the Platform for Advanced Characterisation of Grenoble (PAC-G) [27], [28]. A captured flux of $1.92 \times 10^9 n/cm^2/s$ has been measured by Au foil activation. TENIS beam is a 5x5 cm^2 square. As shown in Figure 3, the flux is very stable in a 2x2 cm^2 square for then decreasing rapidly. The sample was tested initially in the middle of the beam spot where the flux is well characterized. In that position the error rate is so high that in few hours we observed more than 100 SDCs on the smallest convolution configuration (256). Because of the high flux, in center it is not possible to test bigger configurations, though. The high flux of the center position was also problematic as after few hours the devices died, probably due to the gamma rays induced Total Ionizing Dose, and we could no longer have it working. We have then shifted the device at the edge of the beam, moving it with steps of 1mm from 2.7cm to 3.1cm from the center. According to the horizontal beam profile shown in Figure 3, the flux significantly drops starting at 2cm from the center, being approximatively $1.4 \times 10^7 n/cm^2/s$ at 3cm from the beam center. As shown in the Figure, the error rate of the 256 and 1024 convolutions, normalized to the error rate observed at the beam center, follows very well the beam profile measurement. The expected dose rate in SIlicon at TENIS is of about $1,000Gv/h$ from neutron interactions and $250Gv/h$ due to gammas coming directly from the reactor. We do not observe any Total Ionizing Dose effect during our experiments.

To compare EMMA and TENIS results, normally the EMMA flux is normalized with the 25meV equivalent flux (the peak energy at room temperature). The neutron energy spectra of EMMA and TENIS, shown in Figure 4, can be described, in first approximation, as a Maxwell-Boltzmann distribution with a broad peak for thermal neutrons.TENIS, as shown in the plot, has a different spectral contribution of epithermal neutrons than EMMA. To compare EMMA and TENIS results, we convert the "thermal flux" to "25meV-equivalent flux" (25 meV being the peak-energy at room temperature). The "thermal flux", as defined in the JESD89A standard and also as common practice in nuclear physics, is the integrated flux < 0.4 $eV/cm^2/s$. The conversion factor between "thermal flux" and "25meV-equivalent flux" is calculated by integrating the differential flux multiplied by the cross section of B-10 and divided by the cross section of B-10 at 25 meV. The result for EMMA is a factor of 0.71.

All experiments are performed at room temperature, using the standard power and frequency configuration of the Coral TPU. We have tested a total of 4 TPUs.

As a consequence of the Covid-19 pandemic situation,

experiments in the UK were performed remotely, thanks to the tireless and precious help of the ChipIR team in mounting the setup and granting remote access to the researchers in Brazil and Italy. Experiments in Grenoble were performed in person, which gave to the researchers an optimistic feeling for the close future of radiation experiments.

IV. Experimental Results

In this Section we present neutron experiments data obtained irradiating the TPUs with atmospheric and with low energy neutrons. We consider both Silent Data Corruptions (SDCs, i.e., errors at the output) and Detected Unrecoverable Errors (DUEs, i.e., crashes or hangs). We first discuss the reliability of atomic operations, i.e. standard (2D) and depthwise (3D) convolutions and then the reliability of four different neural networks that were trained with multiple datasets for a total of 8 neural networks configurations. All data is reported with 95% confidence intervals, considering a Poisson distribution.

A. Atomic Operations

Aiming to analyze how faults affect the execution of the simplest and most light-weighted operations that the TPU can execute, we run two different types of convolutions: standard and depthwise. Standard convolution stands for normal 2D convolutions while, in depthwise convolution, the input has multiple channels and each one is convolved with its respective kernel separately. In our experiments, the inputs of depthwise convolutions always have three channels (as for the RGB colors) and, because of that, this type is referred as 3D convolutions. We performed tests with squared matrixes of varying sizes as inputs and squared kernels of fixed sizes: 40 for standard convolutions and 20 for the depthwise ones. We choose a kernel size that is both representative (kernel size is normally much smaller than the feature size) but yet sufficient to saturate the TPU computing capabilities (a 40 kernel would exceed the TPU computing capabilities for the 3D convolution).

Figure 5 plots the cross sections (SDCs in blue and DUEs in yellow) for the tested sizes of both convolution types resulting from the high energy neutrons experiments at ChipIR and TENIS facilities. Due to the low error rate at EMMA (more than 5 hours of experiment was needed to observe 1 error), we decide to test the TPU executing convolutions at TENIS, that provides a 3 order of magnitude higher flux, with cold moderation of neutrons. The results for size 256 of the standard convolution algorithm were obtained at TENIS and are highlighted with a different fill pattern in the left side of the graph. Depthwise convolution for 1,250 input cannot be executed on the TPU since it exceeds the device computing capabilities.

As shown in Figure 5, the SDC cross section increases with the size of the convolution input. Intuitively, this is justified as the systolic array becomes more occupied due to the increasing amount of data that needs to be processed. On the contrary, the DUE cross sections does not follow this trend. This should not surprise since, as shown in one of our previous publications [29], DUEs normally have a component that depend exclusively on the hardware and not the software layer. DUEs, then, are biased by the sensitivity of resources that are independent of the executed code (or input size).

From Figure 5, we observe that, for a given input size, depthwise convolutions have higher SDCs cross section when compared to standard convolutions (on average, 179% higher). Also, the cross sections of 3D convolutions increase with the input size at a higher rate than the 2D convolutions. This is, again, related to the fact much more area of the TPU device is used when processing 3D convolutions.

The cross sections for standard convolutions of size 256, that were irradiated with thermal neutrons, have the device positioned in the beam center at TENIS facility. The flux in this position is too high to test bigger configurations. Compared to the Conv 500 values obtained during the experiments with high energy neutrons at ChipIR facility, the cross section at TENIS is about 5 times smaller. This is in line with previous data on thermal VS high energy neutrons obtained in various devices [28], [30]. We recall that the sensitivity to thermal neutrons is strictly related to the amount of Boron-10 used in the device production, which is normally a business sensitive information not available to the public. As observed in [30], the flux of thermal neutrons depends on various factors, including the environmental conditions. It is not possible to provide an expected FIT rate for thermals without knowing the details on the environment surrounding the device.

Figure 6 shows the geometric distribution of the output elements that were corrupted during the experiments with convolutions at ChipIR. When an SDC is detected we download the whole output matrix and identify how many elements in the output are corrupted. When multiple elements are found corrupted we categorize the corruption based on the spatial distribution of the wrong elements. When more than one element is corrupted and these elements sit in the same row or column, we count a *Line* error. When the corrupted elements are distributed in a square (a whole portion of the output matrix is corrupted), we count a *Square* error. When we see multiple corrupted elements that are neither on a Line nor on s Square, we count a *Random* error. It is worth noting that we engineered the experiments (input sizes and flux) not to have more than one neutron generating an error in one execution, since this would be an artifact unlikely for a realistic application. Thus, eventual multiple errors are caused by the spread of the single neutron corruption to multiple operations and not by multiple neutrons corruption.

Regardless of the convolution type, we observe that the distribution is very similar across all sizes. Most of the time, a single element of the output matrix was corrupted. The second most frequent SDC geometry is Square, meaning that the elements corruption occurred within square/rectangular blocks, followed by Random distribution, in which the position of the errors does not match any geometric shape. Finally, element corruptions arranged in a single Line was the least frequent geometric distribution.

The fact that simple corrupted elements is the more common distribution for the TPU is in contrast with what has been

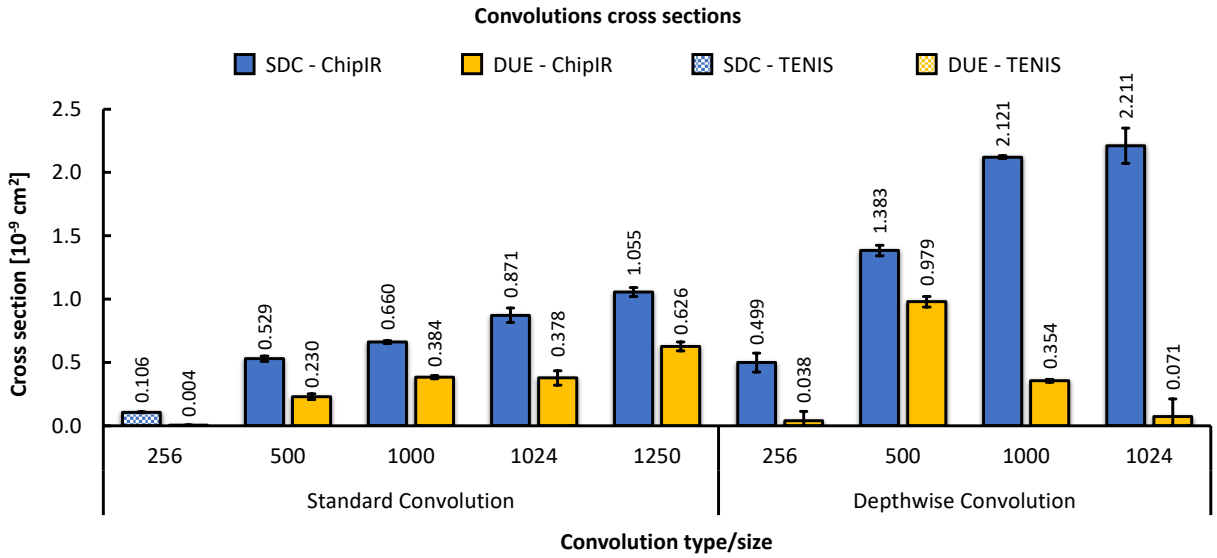

Fig. 5. Cross sections for standard (2D) and depthwise (3D) convolutions, with increasing input sizes, exposed to high energy neutrons at ChipIR and TENIS facilities. Due to the high flux at TENIS we could test only the 256x256 2D configuration in the beam center. For ChipIR we do not report results neither for 256x256 standard convolution as the processed data is too small to provide a sufficient number of errors nor for depthwise 1250x1250 as the input size exceeds the Coral TPU computing capabilities.

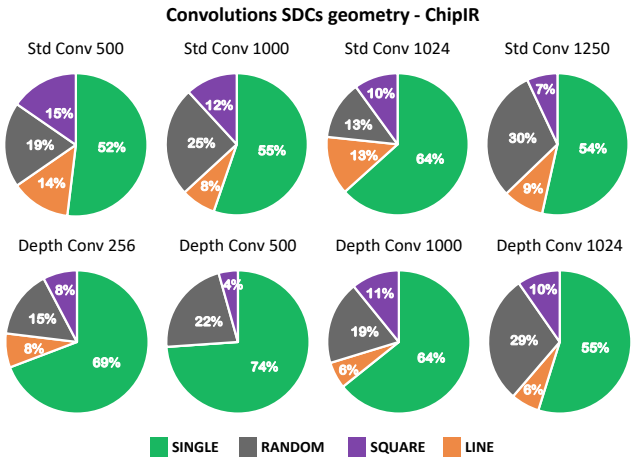

Fig. 6. Geometric distribution of the corrupted elements in the output of convolutions observed at ChipIR (data from TENIS are similar and thus not shown). When multiple output elements are corrupted we categorize the corruption based on the spatial distribution (line, square, or random) of the wrong elements. Most of the time we found just one element corrupted (Single).

observed for Graphics Processing Units (GPUs) [11], [31], [32], for which the majority of the corrupted matrices have multiple corrupted elements. This is due to the different way matrix multiplication is implemented. On GPUs, matrix multiplication is executed as a code, with a sequence of instructions while on the TPU the execution is done with a single instruction in a systolic array. Executing a sequence of instruction might, then, lead to a higher spread of the error in the output. As it has been shown that multiple corrupted elements in the output matrix are the main cause for misdetections or misclassifications in convolutional neural networks [11], the fact that the TPU is less prone to have multiple output errors than GPUs could be a promising results for its reliability in executing CNNs.

Additionally, we have observed that the magnitude of the errors (i.e., how much the corrupted value is different from the expected one) is, overall, very small. The absolute difference between the expected and the corrupt element value is, in fact, exactly *one* (e.g. the expected value is 80 and the corrupted one is 81 or 79) in 91% of the observed SDCs. Please recall that only INT8 operations can be performed on the TPU. Also, when the error magnitude is greater than one, the difference with the corrupted and expected value is a power of 2 (i.e., a single bit flip usually occurs). This happens regardless of the convolution type. Again, this is in contrast with data observed for GPUs, for which the magnitude of the error can be significantly higher (orders of magnitude) [11], [32]. This is another promising result for the TPU reliability in executing CNNs, as a higher error magnitude can have a greater impact on the output value.

B. Neural Networks

With regards to neural networks, we test the reliability of eight different configurations by varying the network architecture, dataset and training procedure, with or without transfer learning, in which the knowledge learned by other model is reused especially to reduce the training time, but also, in most cases, the resulting NN achieves better prediction performance/accuracy).

We leveraged on four NNs models that were trained and made available by Google (Inception V4, ResNet-50, SSD MobileDet, and SSD MobileNet V2). We also retrained MobileNet using two different datasets (a subset of the COCO dataset and a subset of the Oxford-IIIT Pet dataset) with and without applying transfer learning techniques. By retraining the NN models, we want to evaluate: (1) how the number of

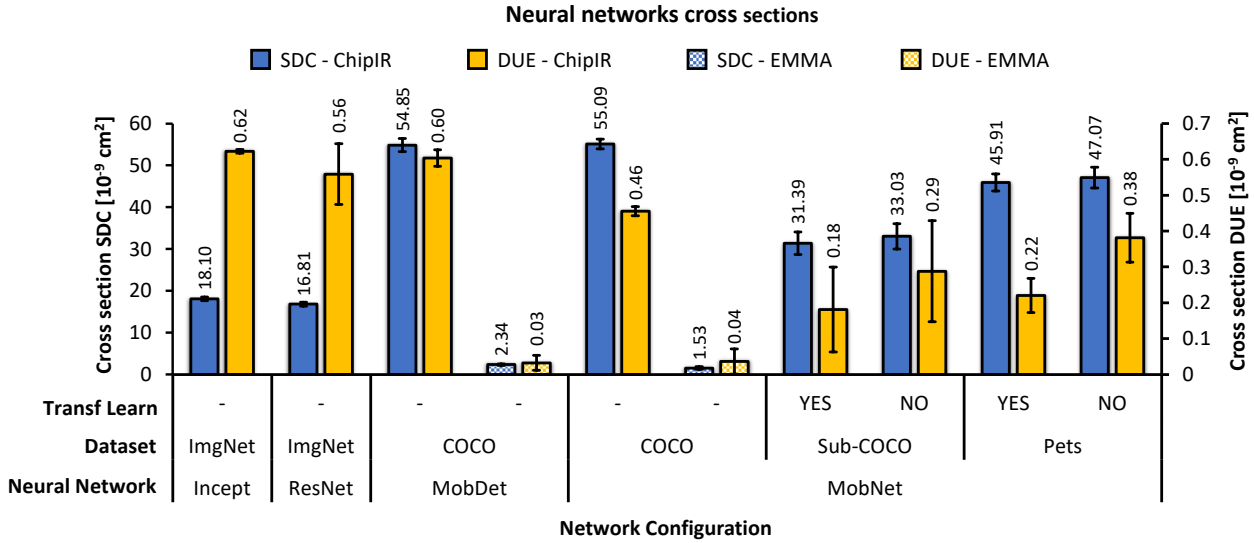

Fig. 7. Cross sections for the eight neural network configurations that were exposed to high energy neutrons at Chip IR facility and to thermal neutrons at EMMA facility. The four configurations on the left side are the original models that Google provide on the Coral Edge TPU official website. The ones disposed on the right side are retrained versions, with and without transfer learning, of the MobileNet with different datasets. The values for SDC cross sections are ploted on the left Y-axis and DUEs on the right.

object classes supported by the NN impacts its reliability, since the datasets used for retraining have much less classes than the original COCO dataset and (2) whether transfer learning has a positive effect into the detection resilience.

Figure 7 plots both SDC cross sections (left Y-axis) and DUE cross sections (right Y-axis) for the eight NN configurations obtained during experiments with high energy neutrons at Chip IR facility and for the two NN configurations tested at EMMA (MobDet and MobNet). Other configurations could not be tested at EMMA due to the low error rate and none of the NNs could be tested at TENIS because the error rate was too high.

At ChipIR, the lowest SDC cross section is measured for ResNet, a similar neural network as the one tested for the NeuroShield in [1]. Considering a $13n/cm^2/h$ flux for atmospheric neutrons at sea level, the $\sim 19 \times 10^9 cm^2$ cross section of the ResNet neural network translates in about 270 FIT, very similar to the 10^2 FIT measured for the NeuroShield.

At EMMA, MobileDet is confirmed to be 50% more likely to experience SDCs than MobileNet. Although the trend is the same, the SDC cross sections are, on average, 25 times smaller than the corresponding values obtained for these two network configurations when exposed to high energy neutrons. From the results plotted in Figure 7, we observe that detection networks are less tolerant than classification ones. This can be justified because, although the classification models are larger and, possibly execute more operations, the detection output is much more complex. For the classification task, the output values simply represent the probability of each object class while, in the detection task, the output is composed of six values for each possibly detected object: its class, its probability and its position (x, y, width, height). The position elements are much more sensitive to the effects of faults and, thus, detection NNs will have higher error rates. This behavior is in accordance with what has been observed in GPUs architectures [11].

Transfer learning (TL) does not seem to have a significant impact on the NN cross section. This technique has shown to decrease the SDC cross section in 2-5% when compared to the analogous configuration without TL. Furthermore, the training process tend to converge much faster with this strategy and, in our case, it reduced the learning time of the NNs in around 50%. TL is a good solution when a quick re-training of the NN is needed, as it is fast but does not impact the error rate.

Our results also confirm that the retraining of MobileNet with the COCO subset (14 classes) lowers the cross sections when compared to the original model trained with the total amount of 90 classes of the original COCO dataset. The same network but trained with the Pets dataset (2 classes) have higher cross sections than the one trained with Sub-COCO, but still smaller than the one obtained for the original with the entire COCO dataset. This trend evinces that, with less classes to be considered, the detection process gets simpler and the cross section is reduced. Therefore, the training of NNs should target the real application needs and include classes of object that are really relevant to the context of the application.

Finally, ResNet and Inception, which are NNs that perform image classification (not detection), have the highest DUEs cross sections. This might be related to the size of the model for these two networks which are 5 to 7 times larger than the MobileNet model. Apart from the retrained networks, which have the lowest value for DUE cross sections, the overall DUE rate is similar among the other NNs which enforces the fact that DUEs are mostly related to the hardware attributes rather than the algorithm.

It is well known that not all SDCs are critical for neural networks execution. Figure 8 shows, for the configurations presented in Figure 7, the percentage of SDCs that critically affect the classification/detection outcome, i.e, SDCs that change the number of detected objects or their classes or even significantly

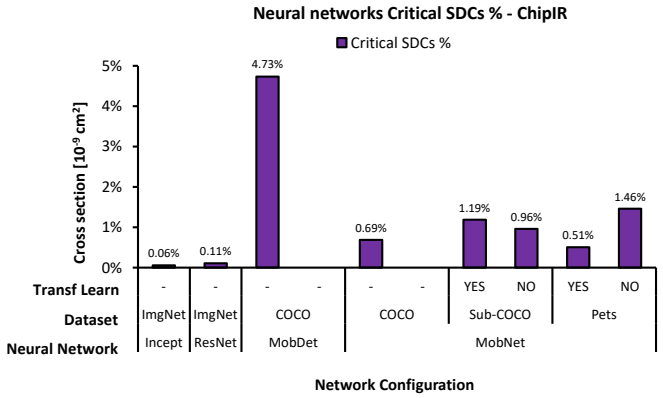

Fig. 8. Percentage of SDCs that critically affected the classification/detection outcome of the neural network configurations that were exposed to high energy neutrons at Chip IR facility.

altered their position (less than 50% of intersection between the expected and the corrupted result).

From Figure 8, it is clear that SDCs in MobileDet tend to be way more critical than the other network architectures. Comparing it to the MobileNet network architecture, which also performs the detection task, MobileDet has less model parameters, a 13% larger input size and a 50% smaller output. The fact that MobileDet has half of the number of output elements makes each one of them twice more significant fot the detection outcome and, therefore, the corruption of a single output value tend to be more critical in MobileDet than MobileNet.

Transfer learning does not seem to have a consistent impact on the criticality of the SDCs. In the case where MobileNet is retrained with the Pets dataset, the application of this technique has shown to decrease the number of critical errors by almost 3 times. On the other hand, when trained with COCO subset, it makes the NN 20% more susceptible to critical SDCs. Further studies are necessary to understand the reasons for this opposite trend. The differences, though, are not very high.

Naturally, SDCs in classification NNs are considerably less critical since only a few values, the highest ones, out of 1,000 output values are indeed relevant to the outcome of the classification process. Therefore, although the SDCs are propagated to the network raw output, most of them do not influence the classification result, as confirmed by our data plot in Figure 8.

V. Conclusions

In this paper we have evaluated the reliability of Google Tensor Processing Units through high energy and thermal neutrons. First, we have understood how neutrons impact the execution of 2D and 3D convolutions, which are the atomic operation for the TPU, of increasing input size. Besides the not surprising linear dependence between the input size and the cross section, we have seen that most neutrons corrupt only one element of the output matrix and the corrupted value is very close to the expected value. Then, we have executed eight different configurations of neural networks on the irradiated TPU. We have seen that detection networks have a much higher error rate than classification networks and that transfer learning does not significantly modify the error rate. Also, the great majority of errors are not critical for the neural network execution, which is strictly related to the fault model observed for convolutions. Finally, the TPU seems more prone to be corrupted by high energy neutrons than by thermal neutrons.

References

[1] S. Blower *et al.*, "Evaluating and mitigating neutrons effects on cots edgeai accelerators," *IEEE Transactions on Nuclear Science*, vol. 68, no. 8, pp. 1719–1726, 2021.

[2] R. M. Brewer *et al.*, "The impact of proton-induced single events on image classification in a neuromorphic computing architecture," *IEEE Transactions on Nuclear Science*, vol. 67, no. 1, pp. 108–115, 2020.

[3] J. Redmon *et al.*, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015. [Online]. Available: http://arxiv.org/abs/1506.02640

[4] M. A. Hanif *et al.*, "Error resilience analysis for systematically employing approximate computing in convolutional neural networks," in *2018 Design, Automation Test in Europe Conference Exhibition (DATE)*, 2018, pp. 913–916.

[5] S. S. Sarwar *et al.*, "Energy efficient neural computing: A study of cross-layer approximations," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 8, no. 4, pp. 796–809, 2018.

[6] V. Mrazek *et al.*, "Design of power-efficient approximate multipliers for approximate artificial neural networks," in *2016 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, 2016, pp. 572–578.

[7] S. Gupta *et al.*, "Deep learning with limited numerical precision," in *Proceedings of the 32nd International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, F. Bach and D. Blei, Eds., vol. 37. Lille, France: PMLR, 07–09 Jul 2015, pp. 1737–1746. [Online]. Available: https://proceedings.mlr.press/v37/gupta15.html

[8] G. Gambardella *et al.*, "Efficient error-tolerant quantized neural network accelerators," *2019 IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, pp. 131–136, Oct 2019. [Online]. Available: http://dx.doi.org/10.1109/DFT.2019.8875314

[9] Q-engineering, "Google Coral Edge TPU explained in depth." Accessed: 2021-08-27. [Online]. Available: https://qengineering.eu/google-corals-tpu-explained.html

[10] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[11] F. F. d. Santos *et al.*, "Analyzing and increasing the reliability of convolutional neural networks on gpus," *IEEE Transactions on Reliability*, vol. 68, no. 2, pp. 663–677, 2019.

[12] A. Bosio *et al.*, "A reliability analysis of a deep neural network," in *2019 IEEE Latin American Test Symposium (LATS)*, 2019, pp. 157–162.

[13] F. Libano, P. Rech, L. Tambara, J. Tonfat, and F. Kastensmidt, "On the reliability of linear regression and pattern recognition feedforward artificial neural networks in fpgas," *IEEE Transactions on Nuclear Science*, vol. 65, no. 1, pp. 288–295, 2018.

[14] F. Fernandes dos Santos *et al.*, "Reliability evaluation of mixed-precision architectures," in *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, Feb 2019, pp. 238–249.

[15] F. Libano *et al.*, "Understanding the impact of quantization, accuracy, and radiation on the reliability of convolutional neural networks on fpgas," *IEEE Transactions on Nuclear Science*, vol. 67, no. 7, pp. 1478–1484, 2020.

[16] C. Szegedy *et al.*, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," in *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, ser. AAAI'17. AAAI Press, 2017, p. 4278–4284.

[17] K. He *et al.*, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[18] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[19] Y. Xiong *et al.*, "MobileDets: Searching for Object Detection Architectures for Mobile Accelerators," *CoRR*, vol. abs/2004.14525, 2020. [Online]. Available: https://arxiv.org/abs/2004.14525

[20] M. Sandler *et al.*, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[21] T. Lin *et al.*, "Microsoft COCO: common objects in context," *CoRR*, vol. abs/1405.0312, 2014. [Online]. Available: http://arxiv.org/abs/1405.0312

[22] O. M Parkhi *et al.*, "The Oxford-IIIT PET DATASET," Accessed: 2021-08-27. [Online]. Available: https://www.robots.ox.ac.uk/ vgg/data/pets/

[23] C. Cazzaniga and C. D. Frost, "Progress of the scientific commissioning of a fast neutron beamline for chip irradiation," *Journal of Physics: Conference Series*, vol. 1021, p. 012037, may 2018. [Online]. Available: https://doi.org/10.1088/1742-6596/1021/1/012037

[24] D. Chiesa *et al.*, "Measurement of the neutron flux at spallation sources using multi-foil activation," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 902, pp. 14–24, 03 2018.

[25] C. Slayman, *JEDEC Standards on Measurement and Reporting of Alpha Particle and Terrestrial Cosmic Ray Induced Soft Errors*. IEEE, 09 2010, vol. 41, pp. 55–76.

[26] C. Cazzaniga *et al.*, "Dosimetry of thermal neutron beamlines at a pulsed spallation source for application to the irradiation of microelectronics," *IEEE Transactions on Nuclear Science*, vol. 68, no. 5, pp. 921–927, 2021.

[27] J. Beaucour *et al.*, "Grenoble large scale facilities for advanced characterisation of microelectronics devices," in *2015 15th European Conference on Radiation and Its Effects on Components and Systems (RADECS)*, 2015, pp. 312–316.

[28] C. Weulersse *et al.*, "Contribution of thermal neutrons to soft error rate," *IEEE Transactions on Nuclear Science*, vol. 65, no. 8, pp. 1851–1857, 2018.

[29] V. Fratin *et al.*, "Code-dependent and architecture-dependent reliability behaviors," in *2018 48th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*, June 2018, pp. 13–26.

[30] D. Oliveira *et al.*, "Thermal neutrons: a possible threat for supercomputer reliability," *The Journal of Supercomputing*, vol. 77, no. 2, pp. 1612–1634, 2021. [Online]. Available: https://doi.org/10.1007/s11227-020-03324-9

[31] P. Rech *et al.*, "An efficient and experimentally tuned software-based hardening strategy for matrix multiplication on gpus," *IEEE Transactions on Nuclear Science*, vol. 60, no. 4, pp. 2797–2804, 2013.

[32] P. M. Basso *et al.*, "Impact of tensor cores and mixed precision on the reliability of matrix multiplication in gpus," *IEEE Transactions on Nuclear Science*, vol. 67, no. 7, pp. 1560–1565, 2020.