Low-High Power Consumption Architectures for Deep Learning Models Applied to Hyperspectral Image Classification

Juan M. Haut, *Student Member, IEEE,* Sergio Bernabé, Mercedes E. Paoletti, *Student Member, IEEE,* Ruben Fernandez-Beltran, Antonio Plaza, *Fellow, IEEE,* and Javier Plaza, *Senior Member, IEEE*

Abstract

Convolutional neural networks (CNNs) have emerged as an excellent tool for remotely sensed hyperspectral image (HSI) classification. Nonetheless, the high computational complexity and energy requirements of these models typically limit their application in on-board remote sensing scenarios. In this context, low-power consumption architectures are promising platforms that may provide acceptable on-board computing capabilities to achieve satisfactory classification results with reduced energy demand. For instance, the new NVIDIA Jetson Tegra TX2 device is an efficient solution for on-board processing applications using deep-learning (DL) approaches. So far, very few efforts have been devoted to exploiting this or other similar computing platforms in on-board remote sensing procedures. This letter explores the use of low-power consumption architectures and DL algorithms for HSI classification. The conducted experimental study reveals that the NVIDIA Jetson Tegra TX2 device offers a good choice in terms of performance, cost and energy consumption for on-board HSI classification tasks.

This work has been supported by Generalitat Valenciana (APOSTD/2017/007), the Spanish Ministry (FPU14/02012-FPU15/02090, ESP2016-79503-C2-2-P), and Junta de Extremadura (GR15005), EU (FEDER) and the Spanish MINECO under Grant TIN 2015-65277-R.

J. M. Haut, M. E. Paoletti, J. Plaza and A. Plaza are with the Hyperspectral Computing Laboratory, Department of Technology of Computers and Communications, Escuela Politécnica, University of Extremadura, PC-10003 Cáceres, Spain. (e-mail:juanmariohaut@unex.es; mpaoletti@unex.es; aplaza@unex.es; jplaza@unex.es). Sergio Bernabé is with the Department of Computer Architecture and Automation. Complutense University, 28040 Madrid, Spain. E-mail: sebernab@ucm.es. R. Fernandez-Beltran is with the Institute of New Imaging Technologies, University Jaume I, 12071 Castellón de la Plana, Spain. (e-mail: rufernan@uji.es).

Index Terms

Hyperspectral image (HSI) classification, deep-learning (DL), low-power consumption architectures, embedded computing.

I. Introduction

The use of miniaturized satellites (SmallSats) is becoming an increasingly popular trend in many of the existing Earth observation programs [1], allowing for a substantial reduction of financial costs and hardware complexity [2]. As a result, this technology has been successfully employed in a wide range of remote sensing applications, such as monitoring of the atmosphere, land cover categorization or mapping of urban areas and the Earth surface [3]. Nonetheless, the increasing demand for extended computing capabilities able to deal with new applications has introduced the need to seek for architectures able to not only increase computing capacity, but also to reduce energy consumption. These requirements may eventually constrain the use of these small devices under highly demanding scenarios, such as the use of deep-learning (DL) techniques for the classification of hyperspectral image (HSI) data [4], [5].

Broadly speaking, HSI collects hundreds of narrow spectral bands in order to simultaneously provide detailed spectral and spatial information, which makes this data especially useful to accurately identify different materials [6], [7]. Many approaches have been proposed to perform HSI classification, however the intrinsic complexity of the HSI domain leads to the fact that only the most advanced convolutional neural networks (CNNs) are able to consistently provide satisfactory results on different remote sensing applications [4], [8]. Furthermore, the selection of efficient computing platform is another critical aspect to take into account, especially when dealing with highly demanding methodologies from a computational point of view. Even though some novel methods pursue to reduce the number of training samples in order to obtain robust classifiers [9], these approaches usually result in computationally demanding models with limited practical application in constrained hardware environments.

On the one hand, commodity clusters [10] and Graphic Processing Unit (GPU) platforms [11] have been traditionally used to process hyperspectral images, but those systems are hardly adaptable to on-board processing requirements which generally introduce strong constraints in terms of energy consumption. On the other hand, field programmable gate array (FPGA) devices [12] offer a good compromise between performance and energy consumption, but they generally require a significant effort from the design and programmability point of view, which may

eventually limit their practical application. In this sense, an attractive alternative is the Tegra GPU architecture which, in the last years, has dominated mobile platforms and embedded devices as Internet of Things (IoT). High rated MPixel/s/Watio, less heat and less space are important keys when facing the on-board processing challenge.

Traditionally, space electronic systems have been highly customized based on the FPGA approach; however, the Tegra architecture is able to provide remarkably higher flexibility while becoming more scalable, affordable and reliable. Even though there are many on-board computing tasks in which Tegra devices may be suitable to process and manage HSI data, it is still necessary to conduct additional research to fully test this hardware applicability. Although there is a handful of jobs that embeds HSI-processing algorithms in efficient platforms [13]–[15], there are very few jobs focused on adapting DL models for remote HSI processing using similar architectures. For instance [16] proposes to integrate a HSI-CNN model (implemented with Caffe framework) into a Jetson TK1 application, reducing the complexity via PCA. However, no implementation details are provided. In this sense, it must be highlighted that although some projects developed by Air Force and NASA experts aim at designing radiation-hardened Tegra hardware for on-board purposes, there are very few research works in the literature aimed at testing the actual performance capability to process HSI data using the most recent DL models for on-board exploitation.

This letter deeply explores, for the first time within the remote sensing research community, the use of DL algorithms over the new NVIDIA Jetson Tegra TX2 low-energy consumption architecture by conducting a comparative study of low-high power consumption hardware applied to HSI classification tasks. The most recent Earth Observation programs work for providing high processing level products which require an increasing demand of ground-segment hardware resources [1]. As a result, studying new alternatives to relieve this work load via on-board low-consumption devices is an interesting option to alleviate ground-segment HSI data computations. In this regard, the target of this work is based on shedding light on the use of the new NVIDIA Jetson Tegra TX2 device for on-board HSI classification when it is compared to other popular hardware alternatives available in regular ground-segment processing units, such as, Intel Xeon and NVIDIA GeForce GTX devices. Initially, sections II-III describe the considered low-high power consumption architectures as well as the DL-based HSI classification models. Then, section IV presents the experimental comparison and highlights the most interesting results. Finally, section V provides interesting conclusions concerning the energy consumption-based

viability of moving specific HSI data computations from the ground-segment resources to on-board platforms via the NVIDIA Jetson Tegra TX2 device.

II. High versus Low-Power Consumption Architectures

Leading manufacturers of high performance computing platforms, such as NVIDIA, launched the Jetson Tegra TX1 device in 2015 as a low power consumption device. This platform was one of the first supercomputers built on a module carrying a Tegra processor from NVIDIA and incorporating an ARM processor. In 2017, NVIDIA announced the new Jetson Tegra TX2 as a compact card design for low power scenarios. This device belongs to the NVIDIA Pascal family, and is an embedded system. The chip features 256 CUDA cores that are based on the same DNA that is featured on the Titan X (Pascal) GPU. The ARM v8 CPU complex comprises two Denver 2 and four A57 cores with a coherent HMP (Heterogeneous Multi-Processor Architecture) geared for multithreading.

In contrast, high power consumption architectures represent now the most widely used choice when power restriction is not necessary. Most of these solutions are based on a workstation featuring a professional Intel Xeon processor in conjunction with one or several NVIDIA GPUs from the Pascal family. Among the main features of the latter is the use of unified memory to solve the limited capacity available on the GPU main memory to process large amounts of data. This mechanism creates a pool of managed memory that is shared between the GPU and the CPU, using a single pointer that is accessible to both the CPU and GPU, bridging the CPU-GPU divide. The data can be read or written from code running on either CPUs or GPUs using calls to *cudaMallocManaged()*. An important aspect is that the Pascal GPU architecture is the first one with hardware support for virtual memory page faulting and migration, via its page migration engine.

In this work, the Jetson Tegra TX2 device (referred to hereinafter as Jetson) is compared against a professional heterogeneous platform (Intel Xeon processor equipped with a GPU NVIDIA GeForce GTX 1080 and referred to hereinafter as Xeon) focused on a detailed comparative study in performance and energy consumption terms. To the best of our knowledge, this kind of analysis has not been previously conducted in the HSI processing literature using DL models for on-board exploitation, and in our opinion it is very important in order to really calibrate the possibility of using low-power consumption platforms for efficient HSI processing in real remote sensing missions.

From a hardware point of view, the main differences between the considered devices are based on the number of CUDA processing cores, memory configuration and thermal design power (TDP). Specifically, the Xeon environment offers over ten times more CUDA cores and streaming multiprocessors (SMs) than the Jetson device. Regarding the memory configuration, we can find some major differences between both the professional (GDDR5X) and Jetson (LPDDR4) platforms. The Xeon platform exhibits higher bandwidth (over 4x) and lower voltage. The 16nm Fin Field-effect transistor (FinFET) technology allows to explore new horizons for discrete memory I/O data rates, from an initial rate between 10 and 12 Gbps to a potential up to 16 Gbps. Moreover, it is possible to reduce the latency gap between local memory and shared internal/external memories through cache prefetching. In the considered professional platform, this technique allows 64B data per memory access to boost execution performance by fetching instructions or data from their original storage in slower memory to a faster local memory before it is actually needed. However, LPDDR4 memory is able to achieve lower memory I/O data rates (between 3.20 and 4.27 Gbps) allowing cache prefetching to 16B. In this way, the power consumption is reduced by lowering the supply voltage (1.1 v) and maintaining an acceptable bandwidth.

Last but not least, power consumption is another important restriction to be considered in on-board processing. In this case, Jetson device presents two performance modes: Max-Q and Max-P. The first one is used on maximum energy efficiency scenarios, where the board TDP sets to 7.5w and Max-P sets to 15w to maximum performance. On the other hand, the professional heterogeneous platform presents an overall TDP of 180w for the NVIDIA GPU and 240w for two Intel sockets considering maximum performance scenarios.

With the aforementioned considerations in mind, we emphasize that the Jetson device offers very encouraging features that make it a competitive platform for on-board processing, with a good trade-off between performance and energy consumption, as compared to other professional platforms.

III. Convolutional Neural Network

To test the performance of the hardware architecture, the spatial CNN model [4] has been adopted. In particular, it is composed by a feature extractor network that receives input data patches of size $d \times d \times 1$, which are obtained from the original HSI cube after applying a PCA-based reduction. The network topology comprises several convolutional layers (CONV),

defined by their corresponding kernel sizes and activation functions (ReLU), in order to learn the non-linearities present in the input data, with the possibility to add a downsampling step performed by pooling layers (POOL). Finally, the extracted features are flattened and sent to the classifier which is implemented as a multilayer perceptron (MLP) with several fully connected layers (FC), some of them equipped with dropout to avoid overfitting. Table I summarizes the topology of CNN models for each HSI dataset.

TABLE I
PROPOSED SPATIAL CNN TOPOLOGY

Layer ID	**Kernel/neurons**	**Ac. func.**	**Pooling**	**Dropout**
CONV1	$32 \times 5 \times 5$	ReLU	2×2	-
CONV2	$64 \times 3 \times 3$	ReLU	2×2	-
FC1	128	ReLU	-	25%
FC2	$n_{classes}$	softmax	-	-

Finally, CNN models have been optimized by using the Adam optimizer with a learning rate of 0.001 (for the Indian Pines datset) and 0.0008 (for the University of Pavia dataset) and 150 epochs. Also, d has been set to 19, 29 and 39 with the aim of testing the computational complexity when different amounts of spatial information have been employed. In this sense, the CNN model needs to fine-tune 54288, 226320 and 422928 parameters for each value of d.

IV. EXPERIMENTS

A. Experimental environment

Two well-known HSIs have been used to perform our experiments. The first one is the know as $145 \times 145 \times 200$ Indian Pines (IP) dataset, captured by the Airborne Visible/Infrared Imaging Spectrometer (AVIRIS) sensor [4] in 1992 over an agricultural area in Northwestern Indiana, comprising 16 different classes. The second dataset is the University of Pavia (UP) scene, acquired by the Reflective Optics System Imaging Spectrometer (ROSIS) sensor [4] over a $610 \times 340 \times 113$ urban area, comprising 9 different classes.

Moreover, two different hardware environments have been considered in this work: i) the **Jetson** (NVIDIA Jetson TX2), which is an ARM GPU environment composed by a dual-core

NVIDIA Denver2 at 2.00 GHz together with a quad-core ARM Cortex-A57 at 2.00 GHz, 8GB 128-bit LPDDR4 and integrated 256-core Pascal GPU at 1300 MHz, and ii) the **Xeon** (multicore heterogeneous system), which is a 2×Intel Xeon E5-2695v3 processors with 14 cores each, running at 2.30 GHz, and 64 GB of DDR3 RAM memory. An NVIDIA GeForce GTX 1080 GPU with 2560 CUDA cores operating at 1772 MHz and dedicated memory of 8 GB.

Regarding the considered software environment, it consists of Debian GNU/Linux 9 and Ubuntu 16.04 as operating systems for both NVIDIA Jetson TX2 and multicore heterogeneous systems, respectively. Tensorflow 1.7 and Compute Unified Device Architecture (CUDA) 8 for GPU functionality.

TABLE II

QUANTITATIVE ASSESSMENT FOR XEON AND JETSON HARDWARE ENVIRONMENTS. IN COLUMNS, WE SHOW THE CONSIDERED BATCH SIZE, THE PATCH SIZE (19×19, 29×29, 39×39) AND THE PERCENTAGE OF TRAINING DATA (5%, 10%, 15%). IN ROWS, WE PROVIDE THE CORRESPONDING OVERALL ACCURACY (%), THE ENERGY CONSUMPTION (WH) AND COMPUTATIONAL TIME (S) FOR EACH EXPERIMENTAL CONFIGURATION.

			19×19						29×29						39×39					
		Batch size	5%		10%		15%		5%		10%		15%		5%		10%		15%	
			XEON	JETSON	XEON	JETSON	XEON	JETSON	XEON	JETSON	XEON	JETSON	XEON	JETSON	XEON	JETSON	XEON	JETSON	XEON	JETSON
INDIAN PINES	Accuracy (%)	25	63.88	64.52	65.53	66.04	67.97	68.86	79.74	79.74	90.69	90.87	93.06	92.9	85.86	87.57	95.00	94.71	97.18	97.59
		50	67.26	68.17	72.51	71.80	74.24	76.16	80.57	80.5	92.78	92.45	95.40	95.48	87.86	87.36	95.21	95.07	97.79	98.07
		100	70.21	69.61	78.85	77.16	80.95	80.69	80.67	79.29	92.78	93.41	96.29	96.53	86.69	87.81	95.23	95.51	97.75	98.31
		200	71.88	71.65	79.89	80.57	83.66	84.74	80.34	80.51	93.45	92.23	96.46	96.38	87.60	87.47	94.78	94.83	98.36	97.95
	Energy (Wh)	25	0.1740	0.0107	0.1749	0.0106	0.1740	0.0105	0.1826	0.0120	0.1817	0.0124	0.1816	0.0124	0.1839	0.0156	0.1919	0.0155	0.1904	0.0156
		50	0.1919	0.0117	0.1944	0.0118	0.1888	0.0120	0.2017	0.0159	0.2038	0.0159	0.1993	0.0160	0.2274	0.0241	0.2290	0.0239	0.2272	0.0238
		100	0.2236	0.0143	0.2212	0.0144	0.2243	0.0144	0.2466	0.0226	0.251	0.0229	0.2517	0.0231	0.3406	0.0448	0.3479	0.0440	0.3494	0.0464
		200	0.2941	0.0196	0.2916	0.0199	0.2898	0.0198	0.3639	0.0416	0.3610	0.0429	0.3638	0.0407	0.5494	0.0898	0.5566	0.0908	0.5604	0.0893
	Time (s)	25	3.98	30.04	4.01	31.06	3.98	30.89	4.04	29.87	4.03	31.94	4.02	31.82	4.11	36.80	4.13	35.65	4.14	35.79
		50	4.24	33.45	4.25	32.63	4.28	33.49	4.34	37.77	4.29	36.72	4.33	37.09	4.59	40.93	4.61	39.61	4.69	39.86
		100	4.90	40.21	4.95	41.12	5.05	41.24	5.19	46.11	5.15	46.01	5.20	46.88	6.03	53.3	6.03	54.2	6.00	50.31
		200	6.27	52.26	6.35	52.49	6.27	49.36	7.25	59.44	7.31	58.42	7.24	60.63	9.17	76.37	9.19	75.43	9.17	75.04
PAVIA UNIVERSITY	Accuracy (%)	25	91.94	90.67	92.62	91.99	92.05	92.80	95.92	96.16	97.89	98.26	99.05	98.72	97.05	97.24	98.99	97.73	98.46	98.45
		50	92.17	92.00	94.11	92.89	94.34	94.94	96.08	96.22	97.75	97.95	99.02	99.11	97.26	97.14	98.57	98.97	99.47	98.75
		100	92.75	93.05	94.86	94.93	96.02	95.28	96.56	96.57	97.85	98.36	99.05	99.18	96.79	96.90	99.14	97.72	99.35	99.33
		200	93.17	92.77	95.95	95.60	96.97	96.63	96.04	96.09	98.06	98.34	99.32	99.31	96.77	96.67	99.07	99.00	99.22	99.30
	Energy (Wh)	25	0.4142	0.0182	0.4104	0.0183	0.4089	0.0185	0.4227	0.0230	0.4147	0.0235	0.4127	0.0247	0.4388	0.0317	0.4374	0.0320	0.4380	0.0320
		50	0.4530	0.0212	0.4463	0.0214	0.4553	0.0219	0.4817	0.0318	0.4756	0.0320	0.4830	0.0322	0.5448	0.0533	0.5281	0.0524	0.5377	0.0545
		100	0.5280	0.0285	0.5267	0.0289	0.5150	0.0288	0.5923	0.0551	0.5910	0.0551	0.5884	0.0549	0.8566	0.1220	0.8607	0.1190	0.8606	0.1154
		200	0.7095	0.0416	0.7124	0.0425	0.6958	0.0427	0.8785	0.1061	0.9032	0.1030	0.8941	0.1011	1.4260	0.2297	1.4169	0.2328	1.4355	0.2348
	Time (s)	25	8.98	63.48	9.01	62.79	9.01	63.23	9.12	60.80	9.12	63.67	9.26	62.20	9.24	68.24	9.17	67.47	9.23	68.79
		50	10.18	66.81	10.14	67.50	10.11	69.63	10.37	75.07	10.24	73.83	10.36	74.73	11.03	81.61	11.09	79.41	11.02	77.84
		100	11.76	88.18	11.73	88.94	11.85	87.62	12.44	100.68	12.41	101.77	12.56	101.24	14.75	105.60	14.81	106.42	14.71	107.64
		200	15.51	111.36	15.47	116.36	15.62	111.95	18.37	133.29	18.34	135.00	18.38	138.27	23.32	180.69	23.50	179.67	23.41	178.97

B. Results

Table II presents the results of our CNN-based classification experiments, conducted on the IP and UP datasets using the two considered hardware environments. In columns, we show the considered input patch size (i.e. 19, 29 and 39), the percentage of training data (i.e. 5%, 10% and

15%) and the corresponding overall accuracy ($\%$) as well as the average energy consumption (Wh) and computational time (s) for Xeon and Jetson environments.

According to the reported quantitative results, it is possible to highlight some important observations. Regarding the classification accuracy, the two considered hardware environments exhibit a similar overall performance. Even though the Xeon environment provides a slightly better average overall accuracy than the Jetson one ($+0.007\%$), the differences between both hardware architectures are always under the standard deviation values, which indicates that these small variations are not statistically relevant and, hence, both environments perform similarly in terms of overall accuracy.

Regarding the energy consumption and processing time metrics, experiments reveal several remarkable differences which deserve to be mentioned. Specifically, the Xeon hardware reports an average energy consumption of 0.4553 Wh whereas the Jetson environment only requires, on average, 0.0452 Wh which makes the former technology 10.06 times more energy demanding than the latter one. When considering the computational time, the Xeon and Jetson environments obtain an average computational time of 9.14 s and 69.79 s respectively. As a result, the Jetson hardware is 7.63 times slower than Xeon, nonetheless it is also 10.06 times more energy-efficient, which generates a positive balance of 2.43 in the energy/performance ratio when considering the Jetson environment.

When analyzing the results in more detail, some interesting points about the tested configurations can be highlighted. More specifically, the obtained quantitative metrics show that the amount of training data does not have a relevant effect on the differences between both hardware environments. That is, increasing the number of training samples from 5% to 10% or 15% does not have an important impact on the computational time, because both Xeon and Jetson environments take advantage of their GPU-based architectures to process the input data, that is, NVIDIA GeForce GTX 1080 and Pascal GPU respectively. However, considering a bigger input patch size affects the two considered hardware configurations in a different way. On the one hand, the Jetson architecture has fewer and slower CPU cores than the Xeon one which logically introduces an unavoidable processing delay as the networks parameters increase. Note that the number of parameters that the CNN model requires to adjust substantially increases with the input patch size, being 4.16 times and 7.79 times the increment when using 29×29 and 39×39 sizes, respectively. On the other hand, the Jetson hardware shares the memory between ARM CPU and Pascal GPU units which makes this hardware less efficient than the Xeon one when

considering very large input spatial sizes, e.g. 39×39, because of the two specific memories present in the Xeon environment. Regarding the considered batch sizes, a similar trend can be observed because Jetson seems to provide a better energy/performance ratio with respect to Xeon when smaller batch sizes are considered.

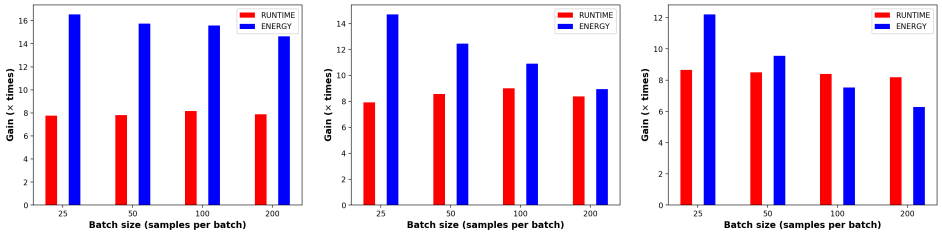

Fig. 1. Runtime and energy differences between Xeon and Jetson environments considering 15% of training data and a spatial size of 19×19 (first column), 29×29 (second column) and 39×39 (third column). In each sub-figure, the red bars represent the runtime improvement provided by Xeon with respect to Jetson (in number of times better) and the blue bars represent the energy savings provided by Jetson with respect to Xeon (also in times better). All experiments have been conducted using the AVIRIS Indian Pines data set.

Fig. 1 displays the runtime and energy differences between the two tested hardware environments in order to highlight the aforementioned points over the Indian Pines dataset. As we can see, the runtime improvements provided by the Xeon environment (red bars) are always lower than the energy consumption savings provided by Jetson (blue bars), except when a 39×39 patch size is considered with 100 and 200 batch sizes. In turn, the Jetson environment is, on average, 7.6 times slower than the Xeon one. The former is about 10 times more energy efficient which clearly reveals its better energy/performance trade-off, especially when not using very large input patch sizes. In the remote sensing HSI classification field, the typical input patch and batch sizes are substantially smaller than the maximum values tested in this work. For instance, a normal patch size value could be 19×19 with 100 batch size (e.g. [4]). As a result, the Jetson hardware environment is shown to be a highly suitable architecture for on-board remote sensing HSI classification, because the energy savings in the acquisition platform are substantially higher than the runtime increase in the ground-segment unit.

Despite the fact that the Xeon environment has shown to obtain a significantly lower computational time than Jetson hardware, it is important to highlight that the latter environment has a much more reduced power consumption while maintaining the classification accuracy which provides an excellent scenario for on-board remote sensing processing tasks. Fig. 2 shows a detailed graphical comparison between the power consumption of both hardware environments

over the Indian Pines dataset in order to better assess the obtained energy results. As we can see, the Jetson energy consumption (displayed in the first row) is substantially lower than the one corresponding to the Xeon configuration (shown in the second row). Besides, the advantage provided by the Jetson architecture becomes especially relevant when considering relatively small batch and patch sizes because of the aforementioned memory limitation of the NVIDIA Jetson Tegra TX2 hardware. With all these considerations in mind, the Jetson environment has shown to provide a competitive advantage in constrained scenarios where power consumption, physical space and financial costs are important decision factors. Precisely, this is the case of remote sensing platforms where this kind of hardware can be an optimal choice to relieve the ground-segment computations when classifying HSI data using relatively simple CNN-based architectures with a constrained number of parameters (i.e. two CNN layers with max-pooling, over a 19×19 input patch size and a batch size up to 100 samples). Consequently, the experimental results and the exhaustive power consumption analysis conducted in this work reveal the viability of integrating the new NVIDIA Jetson TX2 for on-board remote sensing HSI classification.

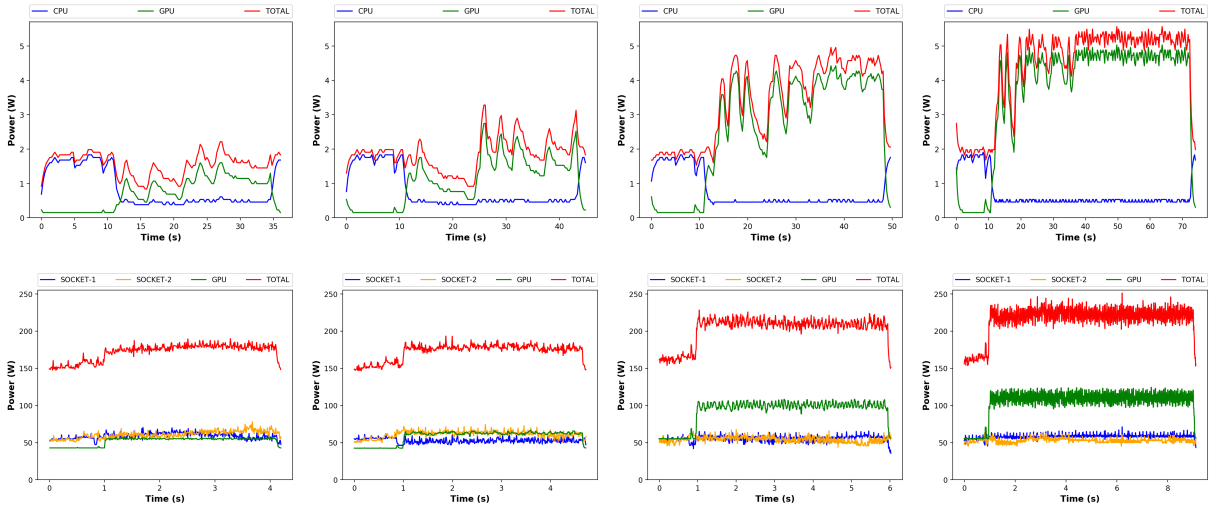

Fig. 2. Energy consumption of Jetson (first row) and Xeon (second row) environments for the four considered batch sizes 25, 50, 100 and 200 (in columns) considering a 39×39 input patch size. Note that each sub-figure shows the used CPU, GPU and total energy consumption (W) and the time (S).

V. Conclusions and Future Lines

This letter studies the possibility of exploiting the new NVIDIA Jetson Tegra TX2 device for on-board hyperspectral image classification in order to relieve ground-segment computations

when generating high-level remote sensing products. Our experimental results, conducted using two different hardware environments and two reference HSI datasets, indicate that the Jetson device provides satisfactory energy/performance results for on-board HSI classification when considering constrained CNN-based architectures. Future work will be focused on analysing other hyperspectral image processing algorithms on additional low-power consumption hardware platforms.

VI. Acknowledgement

The authors would like to thank Dr. Francisco D. Igual for his helpful assistance.

References

[1] A. S. Belward and J. O. Skøien, "Who launched what, when and why; trends in global land-cover observation capacity from civilian earth observation satellites," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 103, pp. 115–128, 2015.

[2] R. Sandau, "Status and trends of small satellite missions for earth observation," *Acta Astronautica*, vol. 66, no. 1-2, pp. 1–12, 2010.

[3] R. Sandau, H.-P. Roeser, A. Valenzuela *et al.*, *Small satellite missions for earth observation*. Springer, 2014.

[4] M. E. Paoletti, J. M. Haut, J. Plaza, and A. Plaza, "A new deep convolutional neural network for fast hyperspectral image classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, 2017.

[5] Y. Chen, Z. Lin, X. Zhao, G. Wang, and Y. Gu, "Deep learning-based classification of hyperspectral data," *IEEE Journal of Selected topics in applied earth observations and remote sensing*, vol. 7, no. 6, pp. 2094–2107, 2014.

[6] J. M. Bioucas-Dias, A. Plaza, G. Camps-Valls, P. Scheunders, N. Nasrabadi, and J. Chanussot, "Hyperspectral remote sensing data analysis and future challenges," *IEEE Geoscience and remote sensing magazine*, vol. 1, no. 2, pp. 6–36, 2013.

[7] R. Fernandez-Beltran, A. Plaza, J. Plaza, and F. Pla, "Hyperspectral unmixing based on dual-depth sparse probabilistic latent semantic analysis," *IEEE Transactions on Geoscience and Remote Sensing*, no. 99, pp. 1–17, 2018.

[8] J. M. Haut, R. Fernandez-Beltran, M. E. Paoletti, J. Plaza, A. Plaza, and F. Pla, "A new deep generative network for unsupervised remote sensing single-image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–19, 2018.

[9] J. M. Haut, M. E. Paoletti, J. Plaza, J. Li, and A. Plaza, "Active learning with convolutional neural networks for hyperspectral image classification using a new bayesian approach," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–22, 2018.

[10] S. Bernabe and A. Plaza, "Commodity cluster-based parallel implementation of an automatic target generation process for hyperspectral image analysis," in *IEEE International Conference on Parallel and Distributed Systems (ICPADS)*, 2011, pp. 1038–1043.

[11] S. Bernabe, G. Botella, G. Martin, M. Prieto-Matias, and A. Plaza, "Parallel implementation of a full hyperspectral unmixing chain using OpenCL," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 10, no. 6, pp. 2452–2461, 2017.

[12] S. Lopez, T. Vladimirova, C. Gonzalez, J. Resano, D. Mozos, and A. Plaza, "The promise of reconfigurable computing for hyperspectral imaging on-board systems: Review and trends," *Proceedings of the IEEE*, vol. 101, no. 3, pp. 698–722, 2013.

[13] L. Riha, J. L. Moigne, and T. El-Ghazawi, "Optimization of selected remote sensing algorithms for many-core architectures," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 9, no. 12, pp. 5576–5587, Dec 2016.

[14] B. Z. B. Y. Z. C. Yuanfeng Wu, Lianru Gao, "Embedded GPU implementation of anomaly detection for hyperspectral images," *Proc.SPIE*, vol. 9646, pp. 9646 – 9646 – 6, 2015.

[15] G. León, J. M. Molero, E. M. Garzón, I. García, A. Plaza, and E. S. Quintana-Ortí, "Exploring the performance–power–energy balance of low-power multicore and manycore architectures for anomaly detection in remote sensing," *The Journal of Supercomputing*, vol. 71, no. 5, pp. 1893–1906, May 2015.

[16] P. H. Randhe, S. S. Durbha, and N. H. Younan, "Embedded high performance computing for on-board hyperspectral image classification," in *2016 8th Workshop on Hyperspectral Image and Signal Processing: Evolution in Remote Sensing (WHISPERS)*, Aug 2016, pp. 1–5.