Conceptual low-cost on-board high performance computing in CubeSat nanosatellites for pattern recognition in Earth's remote sensing

JJ Hernández-Gómez[1], GA Yañez-Casas[2], Alejandro M Torres-Lara[3], C Couder-Castañeda[1], MG Orozco-del-Castillo[4], JC Valdiviezo-Navarro[5], I Medina[1], A Solís-Santomé[1], D Vázquez-Álvarez[3], and PI Chávez-López[6]

[1] Instituto Politécnico Nacional, Centro de Desarrollo Aeroespacial jjhernandezgo@ipn.mx
[2] Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco, Sección de Estudios de Posgrado e Investigación
[3] Instituto Politécnico Nacional, Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco
[4] Tecnológico Nacional de México/I.T. Mérida
[5] Centro de Investigación en Ciencias de Información Geoespacial, Subsede Yucatán
[6] University of California, Riverside, College of Humanities, Arts and Social Sciences, Economics Department

Abstract

Nowadays, remote sensing data taken from artificial satellites require high space communications bandwidths as well as high computational processing burdens due to the vertiginous development and specialisation of on-board payloads specifically designed for remote sensing purposes. Nevertheless, these factors become a severe problem when considering nanosatellites, particularly those based in the CubeSat standard, due to the strong limitations that it imposes in volume, power and mass. Thus, the applications of remote sensing in this class of satellites, widely sought due to their affordable cost and easiness of construction and deployment, are very restricted due to their very limited on-board computer power, notwithstanding their Low Earth Orbits (LEO) which make them ideal for Earth's remote sensing. In this work we present the feasibility of the integration of an NVIDIA GPU of low mass and power as the on-board computer for 1-3U CubeSats. From the remote sensing point of view, we present nine processing-intensive algorithms very commonly used for the processing of remote sensing data which can be executed on-board on this platform. In this sense, we present the performance of these algorithms on the proposed on-board computer with respect with a typical on-board computer for CubeSats (ARM Cortex-A57 MP Core Processor), showing that they have acceleration factors of average of $14.04\times \sim 14.72\times$ in average. This study sets the precedent to perform satellite on-board high performance computing so to widen the remote sensing capabilities of CubeSats.

1   Introduction

On-board processing is fundamental for a space system because it controls the whole operation of the mission's systems. On-board computer subsystems must have the following features: availability, flexibility, processing capacity, data managing as well as reliability [31]. In order to build a processing system with characteristics, it is fundamental to consider the hardware, the instructions set of the architecture (ISA), the communications system's topology, as well as the software to manage and control the satellite [31].

In the beginning of the aerospace industry, on-board processing was required modular in order to tackle the many functions of subsystems of the space mission. IBM systems of the 1960s were successful modular architectures that were used in many NASA missions like the first Saturns [31]. This on-board computer has evolved, and nowadays, several modern processing architectures like microcontrollers, microprocessors and very large scale integrated circuits (VLSI) like FPGA have been used [27]. Along these technological advancements, the internal network to monitor the satellite's tasks has been evolved [14].

Nevertheless, it is a well known fact that the computing power of typical on-board computers is strongly limited by the volume, power and mass of nanosatellites, particularly of those built under the CubeSat standard [9]. Although recent advancements in the development of processors for on-board nanosatellite missions have been achieved, current processors are not able to deliver computing performance for the processing and post-processing of payload's information, as the whole operation of the satellite relies on the on-board computer.

Currently, leading processors used as on-board computers in most nanosatellite missions are microcontrollers or 32-bit based processors as those from the ARM or ATVEL families. For instance, one of the best performance processors that could also be found in a dual-processor configuration is the novel ARM Cortex-A32 MP Core Processor. Nevertheless, these processors have a computing power of the order of 1-4 GFLOP/s. This is the reason why in most on-board systems auxiliary computing units can be found. For instance, an IMU dedicated to process the input of sensors to govern the attitude of the satellite; a common process-demanding task.

In this work, we study the feasibility of usage of an NVIDIA Jetson TX1 embedded system as an on-board system to fully drive the operations of the satellite, which delivers 1 TFLOP/s of processing power in the dedicated graphic unit, which is an unprecedented computing power in micro and nanosatellites. As the algorithms meant to be executed in the GPU must be codified in CUDA C language, this allows the usage of several optimised libraries for GPU, making it a powerful and versatile option for several applications, reducing post processing signal data values and rates. We present performance studies on the acceleration factor of nine very used algorithms in remote sensing implemented in this system. Although our tests were performed with a commercial component, our calculations on the power, size and mass of the considered module show that it is feasible to be implemented in a 1U CubeSat remote sensing mission.

2   Materials and methods

In what follows, information about the used platform as well as about the data processing algorithms considered in this work.

2.1   NVIDIA Jetson TX1

Graphic processing units (GPUs) have been installed alongside multi-core architectures to speed up a huge range of science and engineering application in order to reduce their computing time.

With the introduction of CUDA (Compute Unified Device Architecture), the power of NVIDIA GPUs for scientific and engineering applications was unleashed by extensions to C language [19]. Nevertheless, using low level tools as CUDA C is a complex task because it requires a high knowledge of the GPU architecture to obtain the best speed-up benefits. The integration of GPUs as on-board processing units in space missions, particularly in nanosatellites, has been hindered by three causes: 1) the aforementioned use of low level tools, 2) the high power requirements of GPUs, and 3) the fact that GPUs have not been specifically designed for space environment.

In this sense, and considering the constraints of the CubeSat standard [9], we study the feasibility of the NVIDIA Jetson TX1 embedded system to completely undertake the operations of a 1U CubeSat satellite with remote sensing mission. The technical specifications of the mentioned platform can be observed in Table 1. The power subsystem of a 1U CubeSat typically delivers 5W of DC. In this sense, it might not be able to provide the current required to fully drive the NVIDIA Jetson TX1 module. On the other side, the mechanical specifications (see Table 1) are fully within the CubeSat standards [9], therefore, its integration within a 1U CubeSat mission is fully feasible.

Table 1: Specifications of the NVIDIA Jetson TX1 module.

GPU: 256-core NVIDIA Maxwell	**Electrical:** 5.5 V-19.6 V DC, under 10W
Memory: 4 GB 64-bit LPDDR4	**Mechanical**: 87 mm x 50 mm, 88 g

In what remains of the section, we present nine algorithms that are commonly used in the processing of data arising from remote sensing payloads which we test in the proposed platform to assess the acceleration factor they could obtain, in order to evaluate the feasibility of they to be executed on-board.

2.2   Data Processing Methods

2.2.1   Kriging method (KM)

Kriging method is an advanced interpolation procedure aimed to generate an estimated surface from a set of scattered points. The method fits a mathematical function to the specified points, or to the points within an specific radius to determine the output value for each location. This algorithm assumes that the distance or direction among the sample points indicates an spatial correlation which can be used to generate a surface with adequate variations. Thus, kriging is suitable when there is a directional influence in the spatially correlated distance within the data. In order for kriging method to weigh measured surrounding values to calculate a prediction of a location with no measurements, a geostatistical estimator is used [25].

2.2.2   Empirical Mode Decomposition (EMD)

EMD algorithm produces smooth envelopes defined by local extreme of a sequence and subsequent subtraction of the mean of these envelopes from an initial sequence. It requires to identify all the local extreme connected by cubic spline lines to produce the upper and lower envelopes [11].

This adaptive analysis method is adequate to process non-steady and non-linear time series. The final purpose of EMD is to divide the series in modes, the so called Intrinsic Mode Functions (IMFs) in the time domain, which is the main difference between EMD and other analysis

methods for time series such as Fourier transform, and wavelet decomposition. For further information on EMD, the reader is referred to [8].

2.2.3 Gray Level Co-occurrence Matrix (GLCM)

In Digital Image Processing (DIP), analysis of textures gives important information about the physical features of an object [3]. One successful method to analyse textures is the Gray Level Co-occurrence Matrix (GLCM), which measures the frequency in which different combinations of values in the pixels of an image occur [3]. In this way, the analysis of the GLCM classifies texturally each neighbourhood associated to a pixel within the image [1]. Thus, it allows to recognise and cluster different regions in the figure. This textural approach is easy to compute and is widely used, particularly in the classification of satellite images [3, 17]. Nevertheless, its main drawback is its very intensive computational burden.

2.2.4 High efficiency video compression (HEVC)

One of the main limitations of CubeSats are the impossibility of being equipped with payloads to obtain high resolution real-time video due to their very limited RF communications bandwidths [16]. The evident solution is to implement video compression algorithms to dramatically reduce the size of transmitted information, which is unfeasible because algorithms to highly compress video are of very high processing burden [10].

To be executed on-board a CubeSat mission, these algorithms must provide high visual quality with very high compression levels [10]. Examples of these algorithms are H264 AVC, Dirac, VP8 and HEVC [10]. In this work we asses the performance of HEVC (High Efficiency Video Codding), which provides codding efficiency, easy integration, high resilience to data losses, as well as high implementability in parallel architectures [26]. Recently, a high parallel design of HEVC decoding in heterogeneous architectures has been presented. Experimental results show important acceleration factors of the algorithm in GPU against CPU parallel implementation (44.99$\times$ and 8.44$\times$ for 4K and 8K video respectively) [30].

2.2.5 Encryption algorithm (AES, Advanced Encryption Standard)

Encryption of payload's information is sometimes fundamental when transmitting data, particularly when it comes to reserved information like national security. In this work we assess the performance of the Advanced Encryption Standard (AES) to encrypt high volumes of information using cryptographic keys of 256 bits [15]. Due to the high burden of the algorithm, much work on its parallelisation in different architectures has been performed. In this sense, [5] has shown that the algorithm itself benefits more from parallelisation in GPU than in CPU, mainly because the later ones are optimised for serial processing, the use of large cache sizes as well as their complex instructions sets [5].

2.2.6 Artificial Laser-based Vision (ALV)

Artificial Vision allows the transformation images in to numerical and symbolic data for their further processing. In [24, 16], an on-board artificial vision system able to locate an Earth's reference point by taking a picture of a laser beam pointing directly to the CubeSat is proposed. This, as part of a PAT (Pointing, Acquisition and Tracking) system [23], to establish an optical laser-based high-bandwidth communications system. In order to be able to do so, a very high resolution camera is required which would deliver very large images, [21] such images are to be processed in segmentation [28, 13], parametrisation [6] and interpretation stages [7]. The

outcome is the computation of a centroid, which will allow to set the PAT for tracking and fine pointing [20]. As CubeSats usually orbit in Low Earth Orbits (LEOs), which ranging from 300 to 700 km above the sea level and the sighting time is of the order of 3 to 5 minutes, the ALV algorithm ought to analyse several very high resolution images in real-time to be able locate efficiently the ground station and to establish optical link and transfer information.

2.2.7 Genetic algorithms to enhance features in remote sensing images (GAs)

In the new area of evolutionary computing, Genetic Algorithms (GAs) are optimisation methods based on the natural process of evolution which holds two principles: the survival of the fittest, as well as the generation of new elements with novel characteristics through the crossover operation. Countless are the applications in which GAs have succeeded. In the field of DIP, they have been of aid in the optimisation of the values of parameters for different algorithms. This is the case of the specific design of filters applied to the image processing, which are targeted to enhance determined features of interest. This is a process that can be performed on-board and in real time for pattern recognition of multiple targets in high resolution images acquired by the satellite's payload.

In this work we reproduce the algorithm of [18], which consists in a convolution operation of a specific 2D structure and the original image, where the GA is aimed to find the elements of the 2D kernel which enhances specific features of the image when the convolution filter is applied.

2.2.8 Artificial Neural Networks (ANN) and Principal Component Analysis (PCA) for objects classification in remote sensing images

Objects recognition and classification are some of the most important tasks in DIP. One of the cornerstones of object recognition is the Principal Component Analysis (PCA), an statistical technique developed to dimensional reduction, and later used to identify and classify human faces in images [29]. In this sense, PCA has been successfully applied in several recognition tasks in very different environments.

Nevertheless, with the rise of the field of Artificial Intelligence (AI), the methods of the later have succeeded in boosting the efficiency of classical algorithms as PCA. Particularly, Artificial Neural Networks (ANN) have proved to be efficient in the process of classification of objects. An ANN is a mathematical model inspired in the biological behaviour of neurons and in the brain's structure; thus, it consists of a number of connected nodes through directional connections [12].

It has been noted that a hybrid ANN/PCA algorithm constitutes a good framework to the classification problem. In [4], it has been successfully applied to solve the problem of massive classification of galaxy structures in large image data-sets. In this work we reproduce such paradigm with a convolutional ANN and PCA to perform real-time object recognition and classification in a set of high-resolution images, which would benefit a CubeSat space mission to analyse on-board large sets of images taken with high-resolution payloads.

2.2.9 3D non-steady heat diffusion (HD, Heat Diffusion)

Finally, although not a remote sensing focused algorithm, it is important to note that the satellite heavily relies in the thermal subsystem which protects the satellite from the aggressive temperatures in the space environment. Nevertheless, CubeSats only possess passive thermal subsystems due to the restrictive standard [9]. It would be very beneficial for the mission to

be able to compute the heat diffusion microscopically in the satellite in real time to improve on-board thermal solutions. To do so, it is necessary to solve the dimensional non-steady heat transport equation, $\frac{\partial T}{\partial t} = C \left(\frac{\partial T^2}{\partial x^2} + \frac{\partial T^2}{\partial y^2} + \frac{\partial T^2}{\partial z^2} \right)$, where T is the temperature field. For its discrete solution, we pose a 3D Finite Differences in Time Domain (FDTD) method of second order in space and time which yields an explicit scheme, along with the Courant condition to conserve its stability [22]. The domain is a 1U 3D CubeSat made from Aluminium 6061, and Dirichlet boundary conditions were imposed to simulate space conditions.

3   Results

In order to provide a reliable benchmarking of the performance of the presented algorithms, they were programmed in two ways: in CUDA C and in standard C language, for them to be executed in both the NVIDIA Jetson TX1 embedded system, as well as in a standard CubeSat microprocessor, the 64-bit Quad-core ARM Cortex-A57 MP Core Processor. It must be mentioned that we used this microprocessor because it is incorporated within the NVIDIA Jetson TX1 module. Nevertheless, current CubeSat missions feature ARM processors with less computing power than this one.

   In order to compare the performance of the algorithms in the mentioned platforms, we use the acceleration factor metric which shows the factor by which the algorithm is accelerated in the NVIDIA Jetson TX1 with respect to the ARM Cortex-A57. The results of the performed benchmarking are shown in Figure 1.
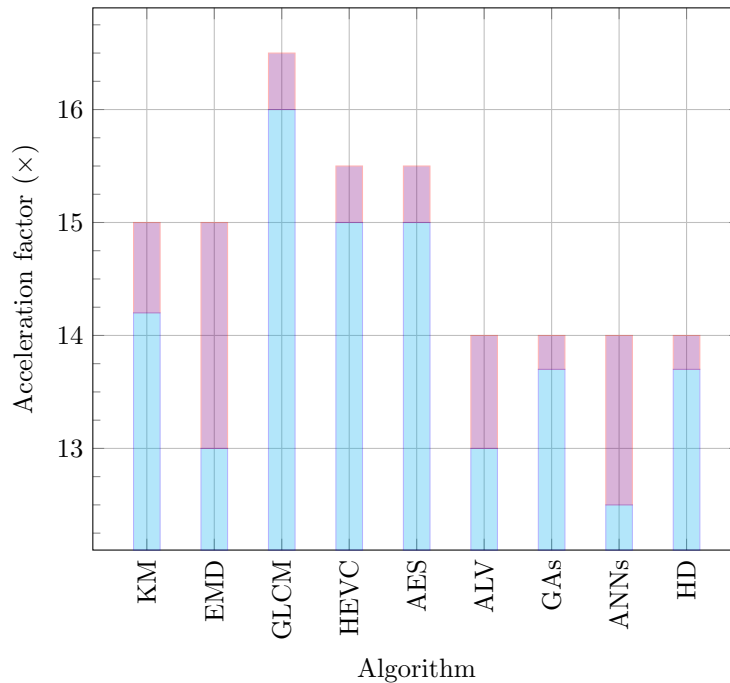

Figure 1: Acceleration factors for each of the algorithms presented in Section 2.2, where the used acronyms can be consulted. Blue bars represent the lowest obtained time while violet ones represent the highest execution time obtained.

From Figure 1 it can be observed that the algorithms are accelerated an average of 14.04×
∼14.72×. It is important to note that the algorithms have a similar acceleration factor, which
implies that despite the fact that these algorithms are very different from each other, they do
not cause a significant change in their execution in the aforementioned platform. In the case of
Heat Diffusion (HD), the low acceleration factor is explained considering that the algorithm is
intensive in the accesses of memory, thus reducing the usage of processor.

On the other hand, as discussed in Section 2.1, the main problem could be the required
current to drive the NVIDIA Jetson TX1 with a 100% burden. In our experiments, the peak
current taken by the module was 8.91 W, so it is fully feasible to power a 2U or 3U CubeSat
with this embedded system (as each unit provides 5W). For a 1U CubeSat mission, the NVIDIA
Jetson TX1 could be installed provided that the computational burden is not too high or that
it is used selectively.

4   Conclusions

In this work we propose to use the NVIDIA Jetson TX1 embedded system as an on-board com-
puter to bring high performance computing to CubeSat nanosatellites whit the aim to perform
on-board real-time computing to process remote sensing payload's data. The mechanical (power
and size) features of the proposed platform satisfy the restrictive CubeSat standard [9], while
the electrical feasibility is demonstrated for 2U and 3U CubeSat missions. It should be noted
that these results coincide with those recently reported by [2] for a TX2 module of the same
manufacturer. the author finds that such module is likewise susceptible to be implemented in
small satellites and CubeSats.

On the side of the performance results, it is important to mention that the platform
was benchmarked against the included-with-the-module ARM Cortex-A57 MPCore Processor,
which outtakes the computing power of common leading on-board processors for CubeSats cur-
rently in use. In this sense, the results show that the computing power offered by the proposed
platform is higher than the average of 14.04× ∼14.72× in acceleration factor obtained with
this processor. This study sets the precedent to open GPU-based high performance parallel
processing in CubeSat satellites for pattern recognition in remote sensing missions.

5   Future work

Besides the implementation of the referred module in a real CubeSat structure and the spatial
validation tests it must undergo, an important direction is to investigate the feasibility and
performance in the recently launched NVIDIA Jetson Nano module. This module is lighter,
smaller, and has lower electrical requirements than the TX1 module, with the drawback of
having less computing power (see Table 2). In this sense, the NVIDIA Jetson Nano module
might pose as the on-board computer ideal for 1U CubeSat missions.

Table 2: Specifications of the NVIDIA Jetson Nano module.

GPU: 128-core NVIDIA Maxwell	**Electrical:** 5 V DC, 5 W-10 W
Memory: 4 GB 64-bit LPDDR4	**Mechanical:** 69.6 mm x 45 mm, 49.7 g

Acknowledgements

References

[1] J. E. Abreu-Torres. Sistema para el reconocimiento semi-automatico de cuerpos de sal en imagenes sismicas. Master's thesis, Universidad Nacional Autonoma de Mexico, Mexico City, 2016.

[2] A.P. Arechiga, A.J. Michaels, and J.T. Black. Onboard image processing for small satellites. In *Proceedings of the IEEE National Aerospace Electronics Conference, NAECON*, volume 2018-July, pages 234–240, 2018.

[3] First Break, editor. *Grey level co-ocurrence matrix and its application to seismic data*, volume 33 of *5*, Paris France, 3 2015. First Break, First Break.

[4] Ricardo Cordero-Chan, Mauricio Gabriel Orozco-del Castillo, Mario Renan Moreno-Sabido, Jorge Javier Hernández-Gómez, Gerardo Cetzal-Balam, and Carlos Couder-Castaneda. Clasificación de galaxias utilizando procesamiento digital de imágenes y redes neuronales artificiales. *Research in Computing Science*, 147(7):319–332, 2018.

[5] TOMOIAGĂ RADU Daniel and STRATULAT Mircea. Aes on gpu using cuda. In *2010 European Conference for the Applied Mathematics & Informatics. World Scientific and Engineering Academy and Society Press*, 2010.

[6] Steven Gao, Keith Clark, Martin Unwin, J Zackrisson, WA Shiroma, JM Akagi, K Maynard, P Garner, Luigi Boccia, Giandomenico Amendola, et al. Antennas for modern small satellites. *IEEE Antennas and Propagation Magazine*, 51(4):40–56, 2009.

[7] Ana González, F Martínez, A Pernía, F Alba, M Castejón, J Ordieres, and E Vergara. Técnicas y algoritmos básicos de visión artificial. *Universidad De La Rioja, Servicio De Publicaciones*, 2006.

[8] Norden E Huang. Review of empirical mode decomposition. In *Wavelet Applications VIII*, volume 4391, pages 71–81. International Society for Optics and Photonics, 2001.

[9] Toorian A. Lan W. Hutputanasin, A. and R. Munakata. CubeSat Design Specification. Technical report, California Polytechnic State University, 2009.

[10] Stamos Katsigiannis, Vasilis Dimitsas, and Dimitris Maroulis. A gpu vs cpu performance evaluation of an experimental video compression algorithm. In *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 1–6. IEEE, 2015.

[11] Donghoh Kim and Hee-Seok Oh. EMD: a package for empirical mode decomposition and Hilbert spectrum. *The R Journal*, 1(1):40–46, 2009.

[12] A. Konar. *Artificial Intelligence and Soft Computing: Behavioral and Cognitive Modeling of the Human Brain*. CRC Press, 1999.

[13] Abdul Halim Lokman, Ping Jack Soh, Saidatul Norlyana Azemi, Herwansyah Lago, Symon K Podilchak, Suramate Chalermwisutkul, Mohd Faizal Jamlos, Azremi Abdullah Al-Hadi, Prayoot Akkaraekthalin, and Steven Gao. A review of antennas for picosatellite applications. *International Journal of Antennas and Propagation*, 2017, 2017.

[14] Christopher J Lowe and Malcolm Macdonald. Space mission resilience with inter-satellite networking. *Reliability Engineering & System Safety*, 193:106608, 2020.

[15] Jianwei Ma, Xiaojun Chen, Rui Xu, and Jinqiao Shi. Implementation and evaluation of different parallel designs of aes using cuda. In *2017 IEEE Second International Conference on Data Science in Cyberspace (DSC)*, pages 606–614. IEEE, 2017.

[16] I Medina, JJ Hernández-Gómez, CR Torres-San Miguel, C Couder-Castañeda, MG Orozco del Castillo, and JI Grageda-Arellano. Artificial vision assisted ground fine pointing system for ex-

perimental optical link for CubeSat communications. *Journal of Physics: Conference Series*, 1221:012063, jun 2019.

[17] Juan Carlos Valdiviezo Navarro, Adan Salazar-Garibay, Alejandro Téllez-Quiñones, Mauricio Orozco-del Castillo, and Alejandra A López-Caloca. Inland water body extraction in complex reliefs from sentinel-1 satellite data. *Journal of Applied Remote Sensing*, 13(1):016524, 2019.

[18] MG Orozco-del Castillo, C Ortiz-Alemán, J Urrutia-Fucugauchi, R Martin, A Rodriguez-Castellanos, and PE Villasenor-Rojas. A genetic algorithm for filter design to enhance features in seismic images. *Geophysical Prospecting*, 62(2):210–222, 2014.

[19] J.D. Owens, D. Luebke, N. Govindaraju, M. Harris, J. Krüger, A.E. Lefohn, and T.J. Purcell. A survey of general-purpose computation on graphics hardware. *Computer Graphics Forum*, 26(1):80–113, 2007.

[20] Scott E Palo. High rate communications systems for cubesats. In *Microwave Symposium (IMS), 2015 IEEE MTT-S International*, pages 1–4. IEEE, 2015.

[21] Daniel Rankin, Daniel D Kekez, Robert E Zee, Freddy M Pranajaya, Daniel G Foisy, and Alexander M Beattie. The canx-2 nanosatellite: expanding the science abilities of nanosatellites. *Acta Astronautica*, 57(2-8):167–174, 2004.

[22] Gerald W Recktenwald. Finite-difference approximations to the heat equation. *Mechanical Engineering*, 10:1–27, 2004.

[23] Kathleen Michelle Riesing. *Development of a pointing, acquisition, and tracking system for a nanosatellite laser communications module*. PhD thesis, Massachusetts Institute of Technology, 2015.

[24] Torres-San Miguel CR Santiago-García LA, Medina I. Desarrollo de un prototipo para localización basado en un algoritmo de visión artifical para cubesats. Technical report, Instituto Politécnico Nacional, Mexico City, 6 2018.

[25] Eric Suarez Gallareta, Jorge J Hernández Gómez, Gerardo Cetzal Balam, Mauricio Orozco del Castillo, Mario Moreno Sabido, and Raúl Alberto Silva Aguilera. Sistema Híbrido Basado en Redes Neuronales Artificiales y Descomposición Modal Empírica para la Evaluación de la Interrelación entre la Irradiancia Solar Total y el Calentamiento Global. *Research in Computing Science*, 147(5):319–332, 2018.

[26] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (hevc) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12):1649–1668, 2012.

[27] Tianyuan Sun, Bingyi Li, Xiaoning Liu, and Yizhuang Xie. An fpga-based balanced and high-efficiency two-dimensional data access technology for real-time spaceborne sar. In *International Conference in Communications, Signal Processing, and Systems*, pages 724–732. Springer, 2018.

[28] Michael Swab, Kyle O'Keefe, and S Scone. Single-frequency ionospheric profiles from the canx-2 nano-satellite. In *Proceedings of ION GNSS*, pages 17–21, 2012.

[29] Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 586–591. IEEE, 1991.

[30] Biao Wang, Diego Felix de Souza, Mauricio Alvarez-Mesa, Chi Ching Chi, Ben Juurlink, Aleksandar Ilić, Nuno Roma, and Leonel Sousa. Highly parallel hevc decoding for heterogeneous systems with cpu and gpu. *Signal Processing: Image Communication*, 62:93–105, 2018.

[31] James R Wertz, David F Everett, and Jeffery J Puschell. *Space mission engineering: the new SMAD*. Microcosm Press, 2011.