INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

RESEARCH ARTICLE

Check for updates

OPEN ACCESS

Received: 02-05-2020

Accepted: 21-05-2020

Published: 18-06-2020

Editor: Dr. Natarjan Gajendran

Citation: Mahendra HN, Mallikarjunaswamy S, Siddesh GK, Komala M, Sharmila N (2020) Evolution of real-time onboard processing and classification of remotely sensed data. Indian Journal of Science and Technology 13(20): 2010-2020. https://doi.org/10.17485/IJST/v13i20.459

*__Corresponding author__.
H N Mahendra

Assistant Professor and Research Scholar, Department of Electronics and Communication, JSS Academy of Technical Education, Bangalore, and Affiliated to Visvesvaraya Technological University, Belagavi, KA, India. Tel.: +91 9740129884
mahendrahn@jssateb.ac.in

Funding: None

Competing Interests: None

Published By Indian Society for Education and Environment (iSee)

Evolution of real-time onboard processing and classification of remotely sensed data

H N Mahendra[1]*, **S Mallikarjunaswamy**[2], **G K Siddesh**[2], **M Komala**[3], **N Sharmila**[4]

1 Assistant Professor and Research Scholar, Department of Electronics and Communication, JSS Academy of Technical Education, Bangalore, and Affiliated to Visvesvaraya Technological University, Belagavi, KA, India. Tel.: +91 9740129884
2 Associate Professor, Department of Electronics and Communication, JSS Academy of Technical Education, Bangalore, and Affiliated to Visvesvaraya Technological University, Belagavi, KA, India
3 Associate Professor, Department of Electronics and Communication, SJB Institute of Technology, Bangalore, and Affiliated to Visvesvaraya Technological University, Belagavi, KA, India
4 Assistant Professor, Department of Electrical and Electronics, RNS Institute of Technology, Bangalore and Affiliated to Visvesvaraya Technological University, Belagavi, KA, India

Abstract

Objectives: To provide a technical review of current hardware architecture, techniques, problems, and practices used for real-time on-board data processing and classification of Remotely Sensed (RS) data. **Method:** The major issues of data processing such as power limitation and downlink bandwidth are considered for analysis. Performance of traditional Central Processing Unit (CPU) and onboard Graphics Processing Unit (GPU), Field Programmable Gate Array (FPGA)based data processing are presented in Table 3. Different hardware architecture used for onboard data classification such as FPGA, Advanced RISC Microcontroller (ARM), and Digital Signal Processor (DSP) based system performance are reported in Tables 5 and 6 respectively. **Findings:** In general satellite data processing, immediate action cannot be taken against natural disasters because of the time taken in processing data at the ground station. Also the downlink bandwidth available between satellite and ground station many not be sufficient to transfer large size of data. One of the solutions to resolve this issue is to process the data onboard, so that data size will be reduced and can be downlink to the ground station for different applications such as urban planning, agriculture, defense/security purposes, biological threat detection, fire tracking on wild land, risk/hazard prevention and also helps to take immediate action during natural disasters. The existing hardware module and its architecture have been studied and concluded with a comparative result. These results aid the researchers to come up with a more optimized design and hardware architecture for data preprocessing and classification.

Keywords: Remote Sensing; pre-processing; classification; field programmable gate array; digital signal processor; graphics processing unit; central processing unit

1 Introduction

Onboard processing of remotely sensed data has an attractive solution for reducing the time of obtaining and processing data at the ground station [1]. In traditional remote sensing the obtained satellite data need to be downlinked to the ground station for processing. At ground station, data will be pre-processed to correct the radiometric, geometric and atmospheric correction to improve the quality of data for different applications such as classification, fire tracks, etc. The procedure for traditional remote sensing is shown in Figure 1.

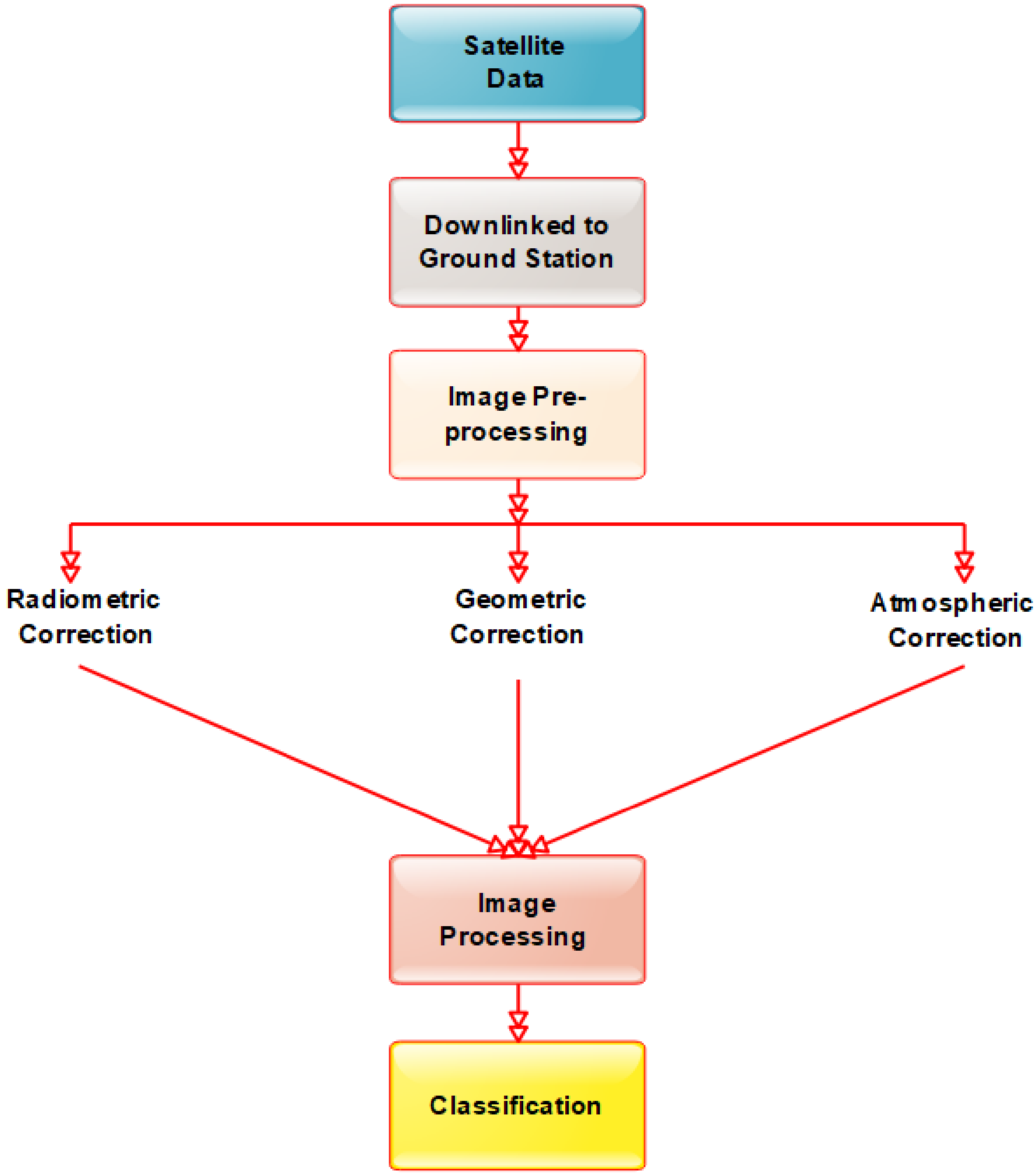

Fig 1. The Procedure of Traditional Remote Sensing

The traditional CPU based processing takes nearly 10min (excluding I/O overhead) to process size of (24,520 pixels x 24,575 pixels) data, this traditional preprocessing will not meet the requirement of real-time preprocessing [2]. The traditional remote sensing will lead to several problems like the requirement of large downlink bandwidth and generates long delays. This procedure cannot apply where we need real-time results. Hence onboard processing is of great interest in the field of remote sensing [3].

In onboard processing the obtained satellite data is processed at satellite platform. The onboard processing can be done in two ways: 1) Satellite data is pre-processed to correct the radiometric, geometric and atmospheric correction, then downlink to ground station for further processing, 2) Satellite data is pre-processed and processed as per the requirement of application such as classification, then downlink the results to ground station for decision making. The procedure for both methods is shown in Figure 2 and Figure 3 respectively.

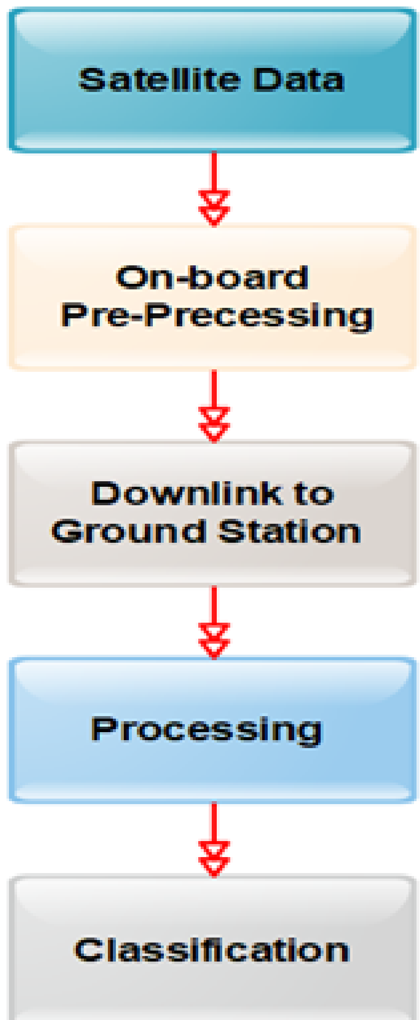

Fig 2. Data Pre-processing at the Satellite Platform

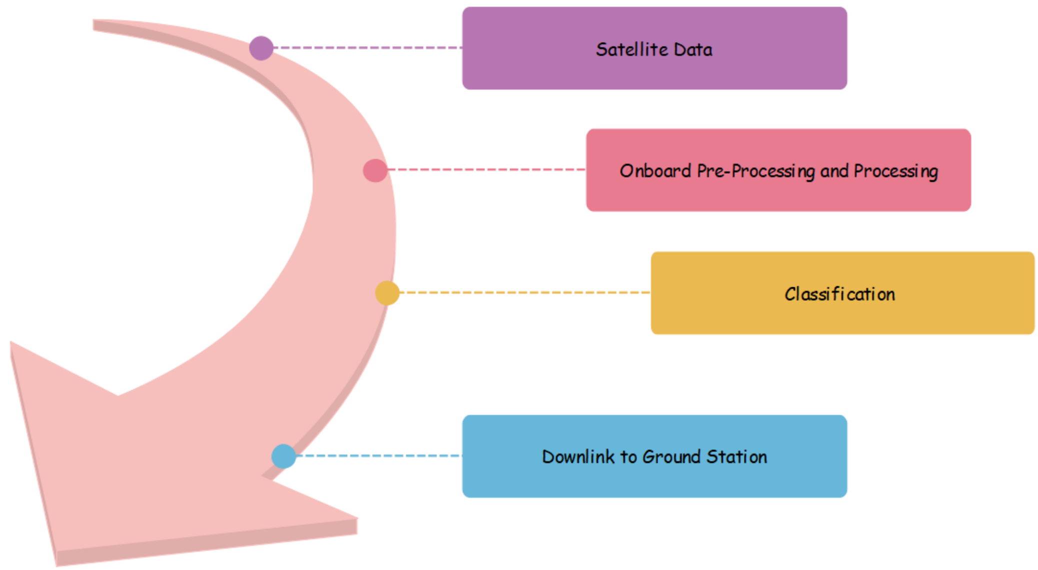

Fig 3. Pre-processing and processing as per the requirement at the Satellite Platform

The rest of this paper is organized as follows: In section 2 importance of onboard data processing is discussed. The performance of traditional and onboard data processing is compared. Also the performance of various hardware architecture available for processing is discussed. In section 3 importance of onboard data, classification is discussed. The performance of traditional and onboard data classification is compared. Also the performance of various hardware architecture available for processing is discussed. The paper concludes with section 4.

2 Data Pre-Processing

Remote sensing technique is widely used for exploring the Earth's surface and it is also helpful in taking precaution measures during natural disasters. The ground station usually demands high-resolution images. Henceforth, high resolution cameras or sensors are incorporated in the satellite. During natural disasters the remote sensing satellites should respond rapidly to send high-resolution images to the ground station in quicker time[(3)].

In the conventional procedures the data is processed at the ground station. So this entire process of capturing the image of ROI (Region of Interest) and sending it to ground station and preprocessing the image takes a long time and this will affect responses to emergencies. To overcome this time lag, on-board processing of the captured image is adopted. Through this, we can minimize the size of the data to be sent to the ground station. Subsequently at the ground station, the data processing flow can be swift and simplified. Consequently ground stations will obtain immediate products to accelerate the decision making against disasters.

The architecture used for onboard pre-processing is reported in[(3)] and this has helped to find different available architecture for pre-processing. In[(3)] the data obtained through high-resolution sensors or cameras of the satellite may contain deformation areas and brightness stripes due to relative motion between the satellite and the Earth, and the defects of the sensors. Therefore the obtained data from the satellite platforms cannot be adopted directly for any specific application. So, preprocessing RS data is a mandatory step to resolve such problems.

Once an image is captured, it undergoes preprocessing, while an image is preprocessed it goes through a processor chain which leads to a required output. The processor in each stage is independent and every processor concentrates on a particular domain. The raw image captured will be stored in the binary form along with ancillary information (orbit, altitude, and imaging time) and the preprocessing chain will have this input, along with this input information the current rigorous modules are also taken. The georeferenced image can be produced by the preprocessing chain, this image can be used for different purposes, like image segmentation and classification. The preprocessing procedure is demonstrated in the flow chart shown in Figure 4.

During preprocessing the raw images are first introduced to Relative Radiation Correction [RRC]. Here the noise which was introduced due to the discrepancy in responses of optical-electronics between different sensors will be removed. Remotely sensed data are susceptible to various distortion like motion blur and atmospheric turbulences. Modulation Transfer Function

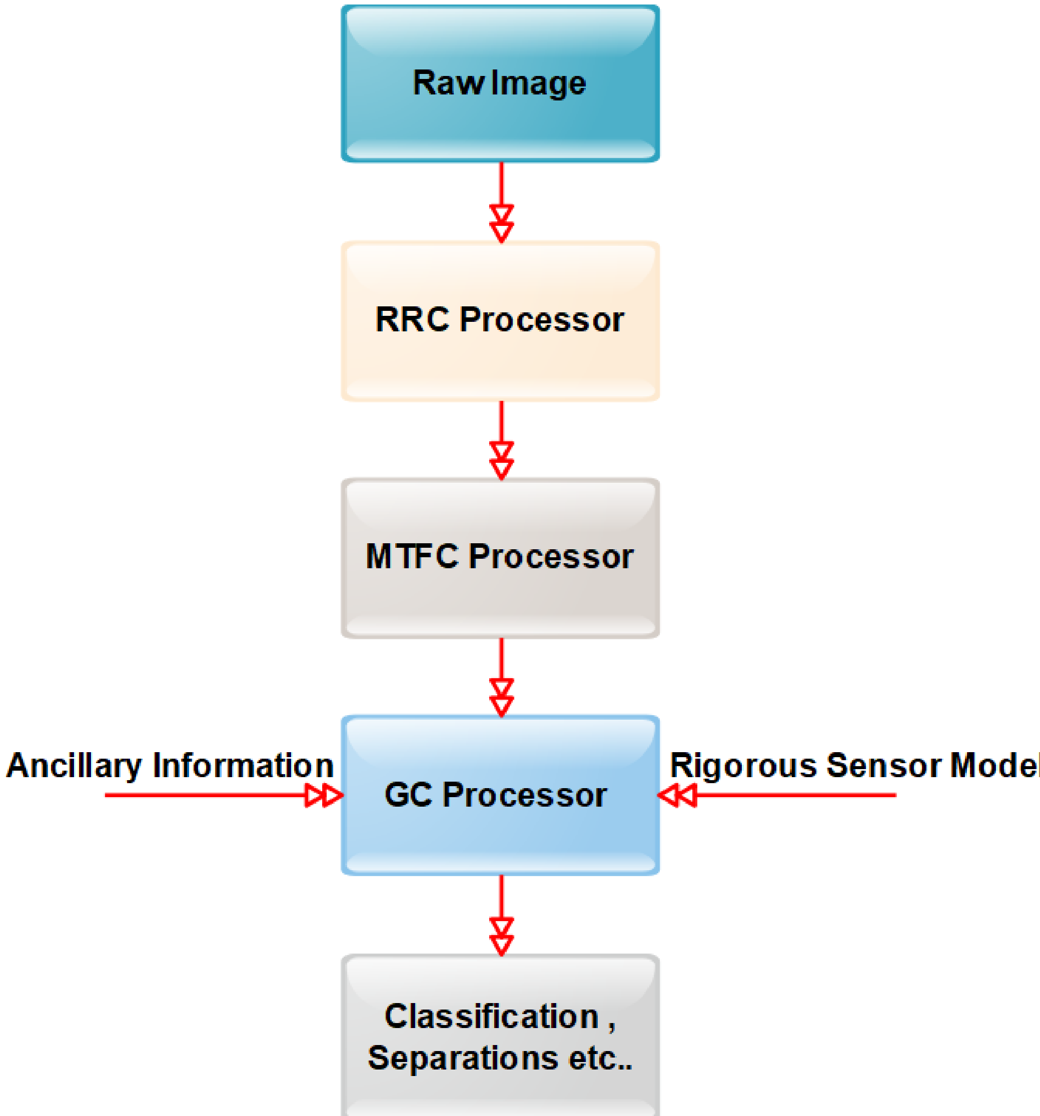

Fig 4. Pre-processing Procedure

Compensation (MTFC) is used as an image restoration technique, so that image quality can be significantly improved and thus help in the image preprocessing and visual interpretation. MTFC processor is used for compensating the image for undesired distortion and degeneration. The RS image is further processed to correct the deformations using Geocorrection (GC) processor that occurs during imaging. There are two types of GC methods non-parametric and parametric, for onboard processing parametric GC model is used because of the availability of the orbital information of the satellite platform.

2.1 Analysis of Remote Sensing Image Preprocessing

The remote sensing data may contain a) system introduced error b) environmentally introduced image degradation, and these errors are radiometrically corrected at the ground station using software such as ENVI, ERDAS Imagine, Geometrical, SAGA GIS.

In [2] describes the time taken to process the 1.12GB size data in different platforms. The RRC, MTFC, and GC run time in the CPU platform are tabulated in Table 1.

Further the author presents real-time approach focusing on the three processors RRC, MTFC, and GC based on GPU pre-

Table 1. The RRC, MTFC and GC run time in CPU platform

Data size (GB)	Processor	Run time of CPU (s)
1.12	RRC	3.64
1.12	MTFC	138.64
1.12	GC	424.21

processing and speedup ratio of the corresponding is tabulated in Table 2.

Table 2. The RRC, MTFC and GC run time in GPU platform and the speedup ratio

Data size (GB)	Processor	Run time of GPU (s)	Speedup ratio
1.12	RRC	0.52	7.01
1.12	MTFC	5.64	24.58
1.12	GC	12.98	32.68

The RS data is obtained at the ground station from the satellite platform. As per ITU, the user downlink data rate is 25.5-27.0 GHz[3]. The ground station always demands high definition images. Therefore it is unlikely to transfer this data in a quick time. Consequently, the decision making against disasters and natural calamities will be delayed. Onboard preprocessing is an effective technique to cut down the delay in RS data preprocessing.

[3] proposed a real-time processing architecture for onboard preprocessing. The preprocessing consists of two parts: - Firstly, the mapping method and hierarchical optimization were used in hardware structure to realize the preprocessing algorithm. Later, to realize a real-time preprocessing, FPGA and DSP based coprocessor were designed. In the coprocessor platform, the RS image was first preprocessed using RRC and the output image was called Level 0, this level 0 image is further preprocessed using a new algorithm developed by the author called as MCCD [Multi charge-coupled device (CCD) Stitching] and the output image of MCCD is called level 1 image. The output of MCCD (Level 1 Image) is again preprocessed during imaging to correct the deformations using the GC processor. The processing time of satellite image on different platforms such as GPU and FPGA/DSP based coprocessor unit are compared with conventional CPU in Table 3.

Table 3. Processing times of RS Image on different platforms. Co-Processor (CP), Central Processing Unit (CPU); and Graphic Processing Unit (GPU)

Platform Model	CPU (sec)(CU)	GPU (sec)(GP)	Co-Processor(sec) (CP)
RRC(R1)	3.64	0.23	0.67
MTFC(M1)	5.64	138.64	NA
MCCDS(M2)	NA	NA	1.67
GC(G1)	424.21	8.49	5.40
Pre-processing time of a satellite image	CU=R1+M1+G1 433.49Sec	GP=R1+M1+G1 147.36 Sec	CP=R1+M2+G1 7.74 Sec

By analyzing the preprocessing time of satellite image in Table 3, CPU based preprocessing at the ground station takes more time to preprocess the satellite image, as the bulk data has to be sent to the Earth station from the satellite platform and the downlink bandwidth from the satellite platform to the Earth station is also a matter of concern here. To overcome the issues of time delay in preprocessing, graphics processors were introduced and the preprocessing time was considerably reduced from 433 sec to 147.36 sec[3]. Then FPGA and DSP based Co-processors were implemented along with a new algorithm of MCCD and this was a real-time onboard preprocessing technique and the preprocessing time dropped down to 7.74 sec.

A detailed comparison of sensor resolution, preprocessing algorithm, implemented platform, performance based on pre-processing time, and power consumption are tabulated in Table 4. This comparison will give an insight to the researchers about the existing algorithms and their efficiency, also it will facilitate the researchers to identify the area of concern and come up with more reliable hardware architecture for onboard RS image preprocessing.

The different types of existing hardware architecture are used to process the remotely sensed data in real-time with their strength and issues are listed in Table 4. This study will help the researcher to choose appropriate hardware to process the data in real-time. In addition, it also helps the researcher to develop new efficient hardware architecture with additional features for real-time data preprocessing.

Table 4. A taxonomy of real-time onboard image processing methods

Ref.	Sensor Resolution	Algorithm used	Platform	Description	Performance Analysis
(2)	Spatial Resolution	RRC/MTFC/GC	CPU/GPU	The processors focused are RRC, MTFC and, GC. The GC and MTFC speedup ratio was very near to theoretical value.	The pre-processing time of GPU is 32 times faster than CPU, but it consumes more power as compared to FPGA. Hence it is not suitable for onboard processing.
(3)	Spatial Resolution	Optimization and mapping method	Co-processor using FPGA and DSP	The FPGA/DSP based co-processor is designed. The pre-processing time of the FPGA/DSP system takes only 7.74 Sec to complete one chain of pre-processing, but CPU and GPU take 433.49 and 147.36 Sec respectively.	The results of the FPGA/DSP based system meet the requirement of real-time processing and the power consumption of FPGA is also less, hence it is suitable for onboard processing.
(4)	Spatial Resolution	Coordinate transformation and Bilinear interpolation	FPGA	FPGA based coordinate transformation and bilinear interpolation are architecture is designed and compared the performance with CPU based system.	The implanted FPGA system is 8 times faster than the CPU based system and power concern is also considered in the design.
(5)	Spectral Resolution	Lossy Compression for Exomars (LCE)	GPU	Hyperspectral and multispectral images are considered for the analysis and the performance of the GPU and the CPU are compared.	The implanted GPU system gives the speedup of up to 15.41 as compared to the CPU system, but power concern is the matter in a GPU based system for onboard processing.
(6)	Spatial Resolution	Pyramidal Blending	GPU	Pyramidal Blending algorithm was implemented on the GPU and the performance of the GPU and CPU are compared.	The implanted GPU system gives the speedup of up to 3.13(s) as compared to the CPU system, but power concern is the matter in a GPU based system for onboard processing.
(7)	Spectral Resolution	(RT-MSVA) Real-time maximum simplex volume algorithm	FPGA	RT-MSVA and RT-FSGA (based on simplex growing) are implemented on FPGA and the performance both are compared. The implanted RT-FSGA and RTMSVA take the processing time of 0.16 and 0.84 sec respectively.	The implemented FPGA based RT-MSVA bit slower than RT-FSGA, but it gives better accuracy in end-member extraction.
(8)	Spectral Resolution	Automatic Cloud Cover Assessment (ACCA) algorithm	Reconfigurable Computer	Reconfigurable computer-based real-time cloud detection system is implemented by considering the flexibility of the microprocessor with FPGA.	The performance of this system is 28 folds higher than the hardware implementation of 2.8Ghz Xenon and the power concern is also considered in the design.
(9)	Spectral Resolution	Data whitening and matched filters detection or classification algorithms	Hardware architecture for Rt^{-1} update for BIP, BIL, BSQ format	With this developed hardware only a small pixel of portion is more enough for detection and classification of images. Even only a small number of the pixels need to be used for data of large homogeneous areas.	The pipeline is introduced to process all formats in parallel, hence it can be used for onboard real-time processing.

Continued on next page

Table 4 continued

Ref.	Sensor Resolution	Algorithm used	Platform	Description	Performance Analysis
[10]	Spectral Resolution	LCMV-Linearly Constrained Minimum Variance	CEM-Constrained Energy Minimization	Real-time processing algorithm is implemented based on LCMV for classification and detection of hyperspectral images.	The implemented LCMV algorithm gives better accuracy for classification and detection in hyperspectral images.
[11]	Spectral Resolution	Sobel edge detector and Gaussian filter image processing algorithms.	FPGA	The designed FPGA system uses DSP slices which contain 18 bit multiplier and 48-bit adder.	The implemented FPGA system, using DSP slice produce accurate results even for a higher number of bits.
[12]	Spectral Resolution	Constrained Energy Minimization (CEM) and Spatial-Spectral Information Extraction (SSIE)	Digital Signal Processor (DSP)	The CEM algorithm is implemented on DSP and SSIE strategy is used to reduce redundant information and improve the Signal to Noise Ratio (SNR).	The implemented system provides a significant speedup up to 90.56, compared to CPU based system.
[13]	Spectral Resolution	Ortho-Rectification Technique	FPGA	The ortho-rectification technique is implemented on FPGA and it speedups the processing of satellite data.	The implemented system process 11,182.3 kilo pixels per second and PC based system process only 2582.9 kilo pixels per second.
[14]	Spectral Resolution	Data-Parallel Process	GPU	The reusable GPU architecture is developed for processing the remote sensing images.	The implemented system helps the designer to reuse the code and speed up the process compared to PC based system.
[15]	Spectral Resolution	Unfixing Chain	GPU/FPGA	The concept of unmixing is used in the implementation of GPU/FPGA.	The implemented GPU takes 17.59s and 24.37s for endmember extraction and abundance estimation respectively. The older FPGA takes 31.23s and 1303.1 s for the same task.

3 Classification of Remotely Sensed Data

Remotely sensed data differ in spectral, radiometric, spatial, and temporal resolutions. The selection of reliable remotely sensed data for image classification is mandatory for analyzing the weaknesses and strengths of different types of sensor data. Classification algorithms and the corresponding hardware are developed based on the user's requirements. Remotely sensed image classification is a complex process and requires consideration of many parameters.

The classification of satellite data may include identification of a suitable classification algorithm, training samples selection, preprocessing of the image, extraction of feature, classification method, post-classification processing, and accuracy assessment. Selection of a number of training sample is critical for classification[16]. When the ROI is complex, the selection of sufficient training samples becomes problematic. Spatial resolution data are of a large volume of mixed pixels, suppose if this data is used for classification, selection of training samples becomes more complicated. Therefore, the training samples must concede the availability of ground reference data, the spatial resolution of the remote-sensing data, and the complexity of ROI[17].

Conventionally data obtained from the satellites were classified using the central processing unit (CPU) at the ground station. Due to the limited bandwidth for data transfer from satellite to ground station and time taken in processing the data, ground station based data classification is not a good option for real-time applications. In[18] and[19], they have designed FPGA (Field Programmable Gate Array) based hardware architecture to classify the hyperspectral images in real-time at satellite platforms. The power consumption is one of the major concerns while classifying the data at the satellite platform. FPGA is the latest available hardware solution for the classification of remotely sensed data in real-time because of its power consumption, size, and weight. Considering these credits currently available radiation tolerant and radiation-hardened FPGAs undoubtedly reflect the best reconfigurable hardware.

3.1 Analysis of Real -Time Classification of Remotely Sensed Data

A different hardware architecture had been considered to find better hardware for real-time classification of RS data. In [18], the design of ZYNQ FPGA using Support Vector Machine (SVM) for real-time classification of RS data and the performance of ZYNQ is compared with the standard design. The RS data of the region of Northwestern Indiana, where the mixed region of agriculture/forestry of size 145 $\times$ 145 pixels was used. Six classes and nine spectral bands were used for identifying and training in datasets.

The first implementation is on the power PC440 processor workstation. The second implementation is on the ARM cortex-A9 embedded system. The third implementation is on the DSP of TMS320C6778. All the reference design results are compared with standard PC based HP XW8600 workstation in Table 5.

Table 5. Performance comparison of different platform

Platform	PC	PPC	ARM	DSP	ZYNQ
T(μs/pixel)	220.3	8516	1321.2	65.8	25.8
E(mJ/pixel)	23.5	83.5	4.3	1.05	0.1
Power (W)	103.8	9.8	3.3	16	3.9
Speedup	8.3	330	1334.5	66.4	26.0

In another study [19], researchers have designed the data flow engine (DFE) on FPGA using support vector machine for real-time classification of RS data and the performance of FPGA is compared with standard PC based design as tabulated in Table 6.

Table 6. Performance Comparison of Different Platform

Platform	PC	DFE	ZYNQ	ARM	DSP	Xeons
T (μs/pixel)	220.3	0.99	25.8	1321.2	65.8	14.1
E(mJ/pixel)	23.5	0.03	0.1	4.3	1.05	1.33
Power (W)	103.8	26.3	3.9	3.3	16	95
Speedup	8.3	1	26.0	1334.5	66.4	14.2

By analyzing the classification time of RS image from the Table 5, classification at the Earth station takes more time to classify the RS image, as the bulk data has to be sent to the Earth station from the satellite platform and the downlink bandwidth from the satellite platform to the earth station is also a matter of concern here.

To overcome the issues of time delay in PC based classification, embedded system (DSP processor) based classification were introduced and the preprocessing time was considerably reduced from 220.3 (μs/pixel) to 65.8 (μs/pixel)) [16]. However, the issues of downlink bandwidth remained unresolved. Then FPGA (DFE and ZNYQ) based system was implemented and this will meet the real-time classification requirements and the classification time was dropped down to 0.99 (μs/pixel DFE) and 25.8 (μs/pixel ZYNQ) respectively.

The DFE based FPGA gets 1334.5x, 14.2x, 66.4x, and 26x increased speed compared to the ARM, Xeon processors, DSP and ZYNQ respectively are tabulated in Table 6. The different types of existing hardware architecture are used in the classification of RS data in real-time with their strength and issues are listed in Table 7. This study will help the researcher to choose appropriate hardware to classify the data in real-time. Also helps the researcher to develop new efficient hardware architecture with additional features for real-time data classification.

Table 7. A Taxonomy of Real-Time Onboard Image Classification Methods

Ref.	Type of RSData	Type of classifier used	Platform	Description	Performance Analysis
(18)	Hyperspectral Image	Support Vector Machine (SVM)	Zynq SoC	The multi classifier SVM is designed and implemented on Zynq SoC, which gives an overall accuracy of 97.8%.	The implemented system increases the speed up to 2.5x ~ 330x with energy-saving of 11x ~ 835x compared to embedded platforms. Hence it is suitable for onboard classification.
(19)	Hyperspectral Image	SVM	FPGA based Dataflow Engine(DFE)	The multi classes SVM is designed based on hamming distance and implemented on FPGA. Also the performance of the system is compared with the various platforms.	The implemented system offers improved speed of 1334.5x, 14.2x, 66.4x, and 26x compared to the ARM, Xeon processors, DSP and ZYNQ respectively.
(20)	Hyperspectral image	Random Forest Classifier	FPGA	The machine learning pixel classifier is implemented on FPGA, which is sensitive to surface materials with texture differences and it can use for onboard classification.	The implemented system helps to find the presence of rocks, layers, and sampling surfaces.
(21)	Hyperspectral image	SVM	–	The performance of various kernel-based classification methods such as neural networks, SVM, and AdaBoost are compared.	SVM provides better results for the classification of remotely sensed data, hence it can be used in hardware development for onboard processing.
(22)	Hyperspectral image	SVM	FPGA	The feed-forward phase SVM is designed and implemented on FPGA. This system reduces the hardware requirement for onboard processing.	The implemented system uses only 167 slices to implement the developed architecture and consumes less power, hence it can use for onboard classification.
(23)	Hyperspectral image	SVM	FPGA	Reconfigurable FPGA is designed to meet the future NASAs onboard processing capabilities and it also has a unique advantage over one time programmable (OTP).	FPGA provides cost-effective prototype environment for the development of future onboard processing.
(24)	Hyperspectral image	Novel Convolutional Neural Networks (CNN)	FPGA	A novel CNN algorithm was designed for hyperspectral image classification and effectively mapped to FPGA for real time processing.	The implemented system 70 times faster than CPU based Intel 8-core and 3 times faster than GPU.
(25)	Spatial Resolution	Neural Networks (NN)	Payload Data Handling System (PDH)	The developed NN algorithm is integrated into PDH for the satellite to classify the data in real-time.	The main aim of the implemented system is recognizing the fire and detection of the hot spots.

4 Conclusion

We studied the evolution of reconfigurable hardware used for real-time onboard processing and classification of RS data. This study helps to understand the role of reconfigurable hardware for remote sensing applications by providing extensive analysis and review the capabilities of FPGA. Based on this study several summarizing statements are formulated.

1. FPGAs are significantly the best choice for real-time onboard processing and classification of remotely sensed data due to their compact size and low power consumption.
2. FPGAs also perform well in signal processing task applications such as processing and compressing.
3. The technological advantage of FPGAs over GPU is, it can be reprogrammable without changing their inner memories.

Hence, the presented work helps to analyze future and current developments in reconfigurable hardware for pre-processing and classification of remotely sensed data in remote sensing applications.

5 Acknowledgement

This research was supported by JSS Academy of Technical Education, Bangalore-560060 and Visvesvaraya Technological University - TEQIP, Jnana sangama, Belagavi-590018 for grant of financial assistance.

References

1) Chen C, Li W, Gao L, Li H, Plaza J. Special issue on advances in real-time image processing for remote sensing. *Journal of Real-Time Image Processing*. 2018;15(3):435–438. Available from: https://dx.doi.org/10.1007/s11554-018-0831-7. doi:10.1007/s11554-018-0831-7.
2) Fang L, Wang M, Li D, Pan J. CPU/GPU near real-time preprocessing for ZY-3 satellite images: Relative radiometric correction, MTF compensation, and geocorrection. Elsevier BV. 2014. Available from: https://dx.doi.org/10.1016/j.isprsjprs.2013.11.010. doi:10.1016/j.isprsjprs.2013.11.010.
3) Qi B, Shi H, Zhuang Y, Chen H, Chen L. On-Board, Real-Time Preprocessing System for Optical Remote-Sensing Imagery. *Sensors*. 2018;18(5):1328–1328. Available from: https://dx.doi.org/10.3390/s18051328. doi:10.3390/s18051328.
4) Liu D, Zhou G, Huang J, Zhang R, Shu L, Zhou X, et al. On-Board Georeferencing Using FPGA-Based Optimized Second-Order Polynomial Equation. *Remote Sensing*. 2019;11(2):124–124. Available from: https://dx.doi.org/10.3390/rs11020124. doi:10.3390/rs11020124.
5) Daga B, Bhute A, Ghatol A. Implementation of Parallel Image Processing Using NVIDIA GPU Framework. In: and others, editor. International Conference on Advances in Computing, Communication and Control. 2011;p. 457–464.
6) Santos L, Magli E, Vitulli R, Lopez JF, Sarmiento R. Highly-Parallel GPU Architecture for Lossy Hyperspectral Image Compression. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*. 2013;6(2):670–681. Available from: https://dx.doi.org/10.1109/jstars.2013.2247975. doi:10.1109/jstars.2013.2247975.
7) Li C, Gao L, Plaza A, Zhang B. FPGA implementation of a maximum simplex volume algorithm for endmember extraction from remotely sensed hyperspectral images. *Journal of Real-Time Image Processing*. 2019;p. 1681–1694. doi:10.1007/s11554-017-0679-2.
8) El-Araby E, El-Ghazawi T, Moigne JL, Irish R. Reconfigurable Processing for Satellite On-Board Automatic Cloud Cover Assessment (ACCA). *Journal of Real-Time Image Processing*. 2019;4(3):245–259. doi:10.1007/s11554-008-0107-8.
9) Du Q, Nekovei R. Fast real-time onboard processing of hyperspectral imagery for detection and classification. *Journal of Real-Time Image Processing*. 2009;4(3):273–286. Available from: https://dx.doi.org/10.1007/s11554-008-0106-9. doi:10.1007/s11554-008-0106-9.
10) Chang CI, Ren H, Chiang SS. Real-time processing algorithms for target detection and classification in hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*. 2001;39(4):760–768. Available from: https://dx.doi.org/10.1109/36.917889. doi:10.1109/36.917889.
11) Wasfy W, Zheng H. General Structure Design for Fast Image Processing Algorithms Based upon FPGA DSP Slice. *Physics Procedia*. 2012;33:690–697. Available from: https://dx.doi.org/10.1016/j.phpro.2012.05.122. doi:10.1016/j.phpro.2012.05.122.
12) Zhang B, Yang W, Gao L, Chen D. Real-time target detection in hyperspectral images based on spatial-spectral information extraction. *EURASIP Journal on Advances in Signal Processing*. 2012;2012(1). Available from: https://dx.doi.org/10.1186/1687-6180-2012-142. doi:10.1186/1687-6180-2012-142.
13) Zhou G, Zhan R, Liu N, Huang J, Zhou X. On-board ortho-rectification for images based on an FPGA. *Remote Sensing*. 2017;9(9). doi:10.3390/rs9090874.
14) Ma Y, Chen L, Liu P, Lu K. Parallel programing templates for remote sensing image processing on GPU architectures: design and implementation. *Computing*. 2016;98(1-2):7–33. Available from: https://dx.doi.org/10.1007/s00607-014-0392-y. doi:10.1007/s00607-014-0392-y.
15) González C, Sánchez S, Paz A, Resano J, Mozos D, Plaza A. Use of FPGA or GPU-based architectures for remotely sensed hyperspectral image processing. Elsevier BV. 2013. Available from: https://dx.doi.org/10.1016/j.vlsi.2012.04.002. doi:10.1016/j.vlsi.2012.04.002.
16) Lu D, Weng Q. A survey of image classification methods and techniques for improving classification performance. *International Journal of Remote Sensing*. 2007;28(5):823–870. Available from: https://dx.doi.org/10.1080/01431160600746456. doi:10.1080/01431160600746456.
17) Toutin T. Review article: Geometric processing of remote sensing images: models, algorithms and methods. *International Journal of Remote Sensing*. 2010;p. 1893–1924.
18) Ma N, Wang S, Ali SM, Cui X, Peng Y. High Efficiency On-Board Hyperspectral Image Classification with Zynq SoC. In: and others, editor. MATEC Web of Conferences. 2016.
19) Wang S, Niu X, Ma N, Luk W, Leong P, Peng Y. A scalable dataflow accelerator for real time onboard hyperspectral image classification. In: and others, editor. International Symposium on Applied Reconfigurable Computing. 2016;p. 105–116.
20) Figueiredo MA, Gloster CS, Stephens M, Graves CA, Nakkar M. Implementation of Multispectral Image Classification on a Remote Adaptive Computer. *VLSI Design*. 2000;10(3):307–319. Available from: https://dx.doi.org/10.1155/2000/31983. doi:10.1155/2000/31983.
21) Camps-Valls G, Tuia D, Bruzzone L, Benediktsson JA. Advances in Hyperspectral Image Classification: Earth Monitoring with Statistical Learning Methods. *IEEE Signal Processing Magazine*. 2014;31(1):45–54. Available from: https://dx.doi.org/10.1109/msp.2013.2279179. doi:10.1109/msp.2013.2279179.
22) Abdul-Halim M, Jallad LB, Mohammed. Hardware Support Vector Machine (SVM) for Satellite On-Board Applications. In: and others, editor. IEEE NASA/ESA conference on adaptive hardware and systems. 2014.
23) Pingree PJ. Advancing NASA's On-Board Processing Capabilities with Reconfigurable FPGA Technologies. In: and others, editor. IEEE International Symposium on Parallel & Distributed Processing. 2010.
24) Liu S, Ringo SW, Chu X, Wang W, Luk. Optimizing CNN-based Hyperspectral Image Classification on FPGAs. *SpringerApplied Reconfigurable Computing*. 2019;p. 17–31. Available from: https://doi.org/10.1007/978-3-030-17227-5_2.
25) Halle W. Thematic Data Processing On-Board the Satellite BIRD. In: International Symposium on Remote Sensing, Proceedings of SPIE. 2001. Available from: http://doi.org/10.1117/12.450683.