Development of GPU-Accelerated Pre-processing Chain for LAPAN-A2 Multispectral Imagery

Kamirul
Satellite Control, Space and Atmospheric Observation, and Remote Sensing Office, National Institute of Aeronautics & Space (LAPAN)
Biak, Indonesia
kamirul@lapan.go.id

Wahyudi Hasbi
Satellite Technology Center, National Institute of Aeronautics & Space (LAPAN)
Bogor, Indonesia
wahyudi.hasbi@lapan.go.id

A. Hadi Syafrudin
Satellite Technology Center, National Institute of Aeronautics & Space (LAPAN)
Bogor, Indonesia
syafrudin.hadi@gmail.com

***Abstract*—This paper reported the development of a newly accelerated chain for pre-processing raw data acquired by LAPAN-A2 multispectral sensor. The acceleration schema was achieved by making full use of *Graphics Processing Unit* (GPU) to handle parallel processing in raw data encoding, image demosaicing, and radiometric-geometric correction stages. Based on the experiment conducted on NVIDIA GeForce GT 1030 GPU card, the developed chain was found capable of resulting level-1 imagery accurately of up to 3.72 times faster than that of similar implementation on Core-i7 8700 CPU. It was also found that the GPU device boosted the demosaicing stage more than 10 times faster. The result of this research demonstrates the utility of the attached multiprocessors of GPU to reduce the execution time of satellite imagery pre-processing.**

Keywords— imagery, GPU, LAPAN-A2, satellite,

I. INTRODUCTION

Optical remote sensing data allow for the mapping and monitoring of Earth's surface at various scales of interest. The capability of this type of data to record up to hundreds of spectral responses of illuminated objects is the main reason to use it in different fields of application and photogrammetry[1]–[3].

According to the capability of optical remote sensing technology, in June 2015, The Indonesian National Institute of Aeronautics and Space (LAPAN) has taken part to demonstrate the ability of a small-sized satellite in performing Earth observation missions by launching an experimental satellite named LAPAN-A2. This satellite was equipped with a high-resolution imaging sensor, SpaceCam C4000. In the present day, from about 650 km of altitude, this payload is capable of producing a RGB multispectral image covering up to 7×7 km^2 of area per single shoot with 3.5m of ground resolution. By using the acquired raw data followed by a sufficient image processing chain, the LAPAN-A2 imageries are regularly used to support various missions, including land cover and infrastructure monitoring, maritime surveillance, and post-disaster analysis [4]–[6].

As the imagery produced by LAPAN-A2 is essential, one of the most important requirements to be fulfilled is the readiness of required data for the analysis. Therefore, a reliable pre-processing capable of converting raw data to an analyzable image becomes a primary concern in developing fast data analysis. Nowadays, as the center of processing for the LAPAN-A2 payload data, Satellite Technology Center of LAPAN has developed an in-house CPU (*Central Processing Unit*)-based chain to perform the required pre-processing. The chain consists of several stages, including raw data decoding, image demosaicing, and radiometric correction. However, as a massive computation occurs during the pre-processing, the duration required to complete all subsequent stages is still not satisfying. Therefore, a new form of chain employing the massive core of multiprocessors attached to GPU (*Graphics Processing Unit)* has been developed to tackle this issue.

The developed GPU-based pre-processing chain for LAPAN-A2 is dedicated to accelerate each stage by adopting the capability of a GPU device to perform parallel computing tasks. The GPU device in which is firstly produced for gaming purposes is equipped up to thousands of cores specified for graphics rendering tasks. In this case, the GPU is only used to handle computation for computer graphics, replacing traditional CPU. In 2001, NVIDIA Corporation started general purposes in GPU computing (GP-GPU) concept to expand the usage of GPU into a wider application. This concept enables a GPU to be programmed directly in addressing scientific computing needs and facilitating real-time processing and analysis of enormous datasets. In the field of remote sensing, there are found numerous successful implementation of GPU, including image classification on hyperspectral imagery, automatic feature and road extraction, noise removal, and near real-time processing and geo-referencing [7]–[10].

In our implementation, there are three major tasks migrated from CPU to GPU platform, i.e., raw data decoding, image demosaicing, and radiometric correction. The raw data decoding mainly consists of byte re-ordering and line header detection stages. The byte re-ordering is required to restore decoded bytes into their correct arrangement, while line header detection is intended to locate uniquely-added markers as the sign of initial pixel line in the image. As the SpaceCam acquired image in the form of color filter array (CFA), an image demosaicing technique is required to reconstruct RGB-colored from incomplete color samples output of the camera device. Finally, in order to improve the quality of the resulted image, a radiometric correction stage consisted of stripe noise removal and line drop correction has also been performed by employing spatial (median and mean) filters [11], [12]. At the end of the development stage, we tested the performance of the developed chain by comparing its execution time to those required by the existing CPU-based implementation.

The aim of this research is to report the development and to assess the performance of a GPU-accelerated image pre-processing chain for LAPAN-A2 imagery system. Moreover, in this paper, we have also delivered an analysis of the advantages of using GPU. Furthermore, we have also clarified limitations associated with the GPU technologies in which should be considered by the developers before implementing a similar approach.

The rest of the paper is organized as follows: Firstly, in Section II, we introduced the stages required to produce imagery from LAPAN-A2 raw data. Secondly, in Section III

978-1-7281-6907-1/20/$31.00 ©2020 IEEE

we presented a detailed explanation of proposed GPU-based implementation, including the setup of the experiment used for assessing its performance. Finally, we provided the results and analysis in Section IV followed by a conclusion in Section V.

II. MATERIAL AND METHOD

A. Image Processing Stages for LAPAN-A2 Raw Data

In order to produce a level-1 (L1) imagery from LAPAN-A2 raw data, three main stages are needed to be done; raw data decoding, image demosaicing, and radiometric correction.

1) Raw Data Encoding

The first stage is raw data encoding. In this step, the bytes of raw data will be re-arranged in order to restore their correct position. Then, a line header detection is applied in order to extract a line of pixels to be used in reconstructing a frame of image.

2) Image Demosaicing

The second stage is the image demosaicing. As the SpaceCam device encoded the acquired image of color filter array (CFA), the demosaicing technique is required to reconstruct the RGB-colored image. In the case of LAPAN-A2, the image reconstructed using Bayer filter has alternating red (R) and green (G) filters for odd rows and alternating green (G) and blue (B) filters for even rows. The Fig. 1 below shows an image reconstructed after performing demosaicing process on LAPAN-A2 decoded raw data.

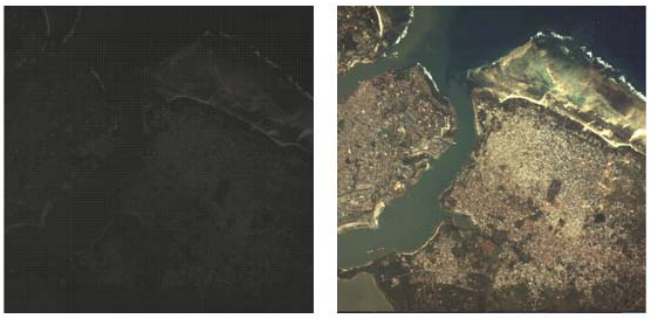

Fig 1. (A) Image of Mombasa taken from LAPAN-A2 SpaceCam and (B) is reconstructed RGB image using 'BGGR' array CFA filter.

3) Radiometric Correction

The last stage is the radiometric correction. There are two types of radiometric correction implemented on the chain, i.e., stripe noise removal and line drop correction. The stripe noise found on the resulted image is caused by differences in the response of detectors, calibration error, and so on. The other one, line-drop effect, appears as raw data contaminated by byte-shifting is fed into previous stages. This type of error is mainly caused by unavoidable errors during data transmission, including attitude error of satellite, frequency interference, weather condition, and the existence of physical blockage around the ground station. All of these issues may corrupt the amount of data; hence the partial bit of data might be lost resulting in a line-drop noise stretching horizontally on the image line,i.e., line-drop. The detail of stripe noise and line-drop removal can be found in [11], [12]. Fig. 2 shows the successful implementation of stripe noise and line-drop correction taken from the mentioned references.

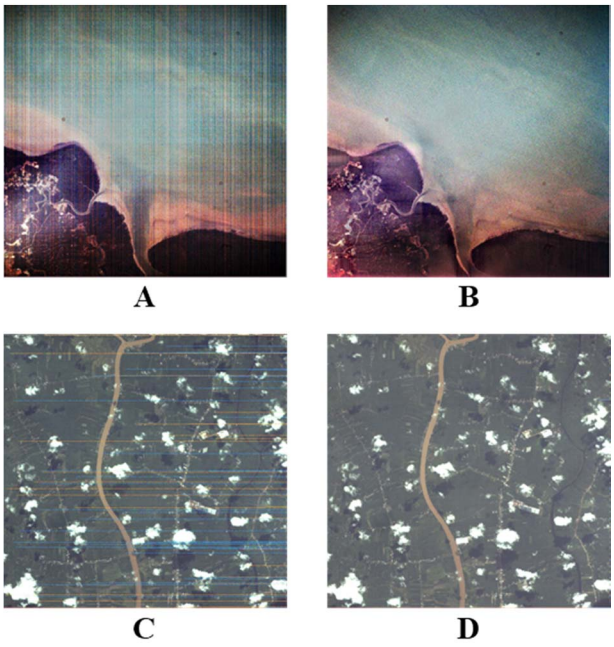

Fig 2. The results of image correction : (A) image disturbed by stripe noise; (B) result of stripe noise removal; (C) image disturbed by line-drop effect; and (D) result after applying line-drop correction [11], [12].

B. Proposed GPU-based Processing Chain

In this section, we provided a detailed explanation on GPU-based image pre-processing chain implemented on LAPAN-A2 raw data.

The developed GPU-based pre-processing chain is the upgraded version of the existing one in which has been designed to use the CPU as its standard computing processor. The new chain reported in this paper is intended to accelerate the pre-processing duration so that the L1 imagery can be produced quickly.

The GPUs could approach an extreme computing speed because they are equipped with streaming multiprocessors. Each multiprocessor consists of hundreds of specific CUDA cores, the smallest processing unit inside the GPU in which could be operated simultaneously in a parallel manner. This type of core is specifically intended to perform an extreme math operation with a huge amount of arguments per instruction. However, for the normal type of instructions, like loops, stack operations, memory operations, especially random access to a huge amount of memory, input-output (IO) operations, math instructions with only one or two or three arguments, the situation changes very much. Therefore, in our developed chain, we actually tried to employ the capability of GPU to boost the math calculations required in the pre-processing chain. The remaining operations are still handled by the CPU.

While working on CPU-based algorithm, raw data stored in computer storage will be first loaded into *Dynam*ic *Random Access Memory* (DRAM), or host memory. Traditionally, all of the computation processes are run by CPU, and the data associated with the computation are temporarily stored in DRAM. However, in GPU programming, a different schema is required in order to access the GPU multiprocessor as the default processor. The simplified version of the proposed GPU-accelerated pre-processing is shown in Fig. 3.

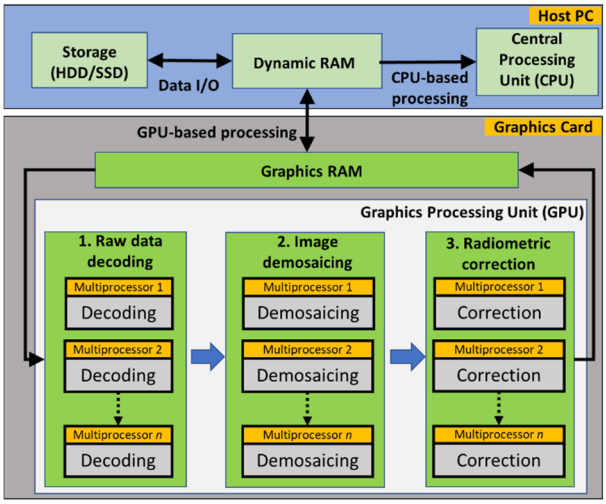

Fig 3. Block of proposed GPU-accelerated pre-processing for LAPAN-A2 imagery

In order to use the GPU multiprocessor inside the graphics card for the processing purpose, the raw data must be supplied to the RAM of the graphics card, i.e., device memory. Basically, the graphics RAM (device memory) has the same functionality as DRAM. However, this type of memory is specifically built for the use of GPU multiprocessor and can be accessed only by GPU during the processing stage. If the required data has successfully resided on the device memory, the subsequent processing stages can be performed by making the full use of GPU multiprocessor. All of these stages are computed in a parallel manner inside the multiprocessor so that the computation can be executed faster than that of the traditional computing method.

C. *Experiment Setup and Evaluation Criteria*

In order to assess the performance of the developed chain, we have conducted an experiment to measure the quality of resulted images and the acceleration achieved by implementing the GPU. The detail of the experiment setup and devices used in this work is provided in Table I.

TABLE I. EXPERIMENT SETUP AND MATERIAL

Data / Device	Specification
Host PC	**Environment** : Windows 10 64-bit **Processor** : Intel Core-i7 8700 (6 cores) **DRAM capacity**: 2 GB
Graphics Card	**Name** : NVIDIA GeForce GT 1030 **Total multiprocessor** : 3 **RAM capacity** : 1 GB **CUDA version** : 10.1
Raw data	151 MB of LAPAN-A2 SpaceCam raw data acquired on July 19, 2018

To measure the quality of images resulted by the developed GPU-based chain, we compared the images to those resulted by the CPU. We then employed the structural similarity index (*SSIM*), a common metric used for measuring the similarity between two images. The *SSIM* value between images x and y can be calculated using Eq. (1).

$$SSIM(x,y) = \frac{(2\mu_x\mu_x)(2\mu_{xy}+c_2)}{(\mu_x^2+\mu_y^2+c_1)(\mu_x^2+\mu_y^2+c_2)} \quad (1)$$

The μ_x, μ_y, σ_x, σ_y, and σ_{xy} respectively refer to the averaged value of x, averaged value of y, variance of x, variance of y, and covariance of x and y. The c_1 and c_2 are two variables used to stabilize the division of weak denominator defined as $(k_1L)^2$ and $(k_2L)^2$. The value of k_1 and k_2 are 0.01 and 0.03, while L represents the dynamic range of the pixel values in which is sets to 2^N-1, with N is the number of bits on the image[13]. The *SSIM* value ranges from 0 to 1, with the value equals to 1 represent a perfect similarity between two assessed images.

To measure the acceleration, we compared the processing time required by the GPU and CPU-based chain to produce L1 images. The term "acceleration" denoted by A to represents the ratio between execution time needed by the CPU-based processing, t_{CPU}, and those required by the GPU-based version, t_{GPU}.

$$A = \frac{t_{CPU}}{t_{GPU}} \quad (2)$$

The value of A larger than 1 is expected for an accelerated system. In our work, the measurement of A ratio is carried out for each particular stage and for overall stages in the form of accumulative duration. As a result, we can analyze the advantage of using GPU for a certain stage of pre-processing and for overall stages seen from the duration of execution.

III. RESULTS AND DISCUSSION

As mentioned at the beginning of the paper, the developed GPU-based pre-processing chain for LAPAN-A2 imagery is an updated version of the existing chain. This new chain has been built in MATLAB platform equipped with a user interface for a better user experience. Fig.4 shows the integration of the new chain to the existing user interface.

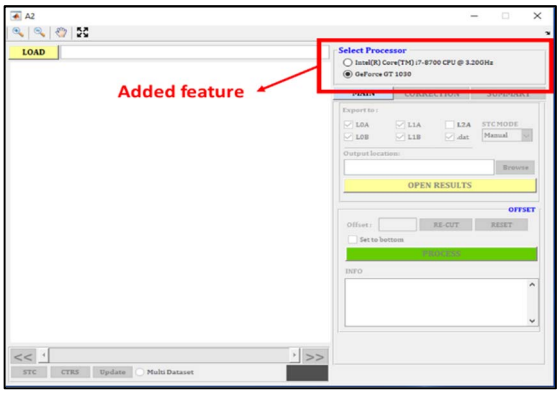

Fig 4. Screenshot of newly-added GPU-based pre-processing chain.

The inclusion of GPU-based chain gives the operator the option whether to perform the processing on GPU or CPU. Moreover, this option enables the PC did not have GPU device to perform the processing in traditional CPU.

Based on the experiment conducted on the CPU and GPU devices listed on Table I using 21 pairs of images, we have measured and found that the averaged *SSIM* values are exactly equal to 1. This result indicates that the GPU-based chain is capable of resulting accurate results as compared to the results of traditional CPU. Moreover, based on the measurement of execution time for every stage of pre-processing, the acceleration factor, A, is collected. In Fig. 5, we provided the comparison of durations required to produce 21 L1 images in GPU and CPU devices, respectively.

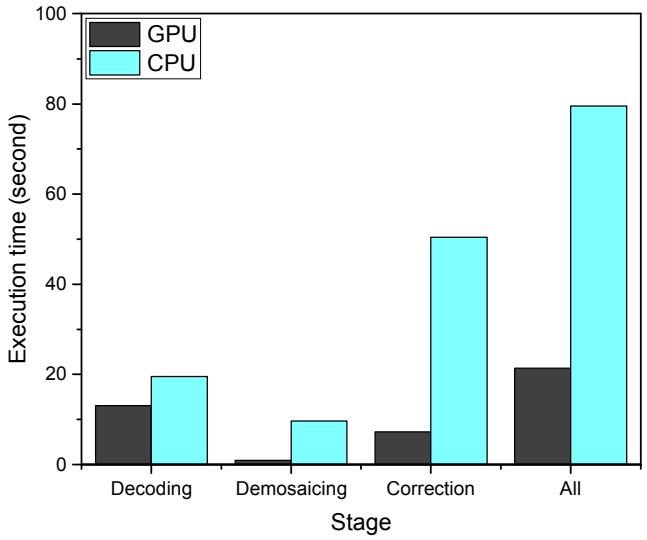

Fig 5. Execution time of CPU and GPU-based pre-processing chain.

Based on Fig. 5, it can be seen that, in general, the radiometric correction stage took the largest portion of pre-processing time in the CPU-based chain. However, in the GPU-based chain, the raw data decoding stage consumes a longer duration than other stages. This phenomenon happened since the GPU-based raw data decoding initiated by data transmission step from host memory of CPU to device memory of GPU. This extra step causes the GPU to consume a larger amount of time than that of the CPU in the decoding stage. To analyze the acceleration resulted by implementing the GPU, in Fig. 6, we provided the acceleration factor gathered from the experiment.

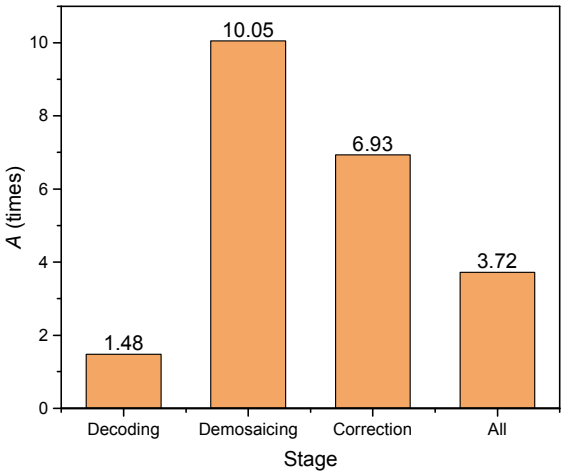

Fig 6. Comparison of acceleration factors of each pre-processing stage.

According to Fig. 6, the GPU device is capable of demonstrating its capacity as a beneficial accelerator in handling the pre-processing stage of LAPAN-A2 imagery. It is also shown that the highest acceleration is achieved as the GPU is used for demosaicing with a value of *A* more than 10. On the contrary, the raw data decoding stage can be accelerated only for about 1.48 from its traditional CPU implementation. As explained in the previous paragraph, this stage consists of a data copying task from host to device memory, which is an unfavorable schema for GPU-based programming. The same phenomenon is also experienced by the radiometric correction stage. In this step, the program has to migrate back the result of computation from GPU memory to CPU memory. Therefore, the raw data decoding and image correction stages are not as fast as the correction stage.

Finally, even the GPU is proven capable of boosting the pre-processing stage of LAPAN-A2 imagery; there is still a disadvantage associated with it. The GPU-based programming generally produces a slow performance as the amount of data to be processed larger than the available memory on GPU. As a result, the transmission of a small chunk of data from the host to GPU memory has to be performed frequently to avoid excessive memory usage. Using a larger GPU memory size is another option to minimize the repetitive copying process between the memory devices. This type of approach can be done by integrating more than one GPU card to perform the same task.

IV. CONCLUSIONS

In this work, a GPU-based chain used for pre-processing LAPAN-A2 imagery has been successfully developed and assessed. The chain has been integrated into the existing chain and can be used as a tool in resulting level-1 (L1) imageries. Based on the experiment, the GPU-based chain is found capable of producing level-1 (L1) accurately with *SSIM* value equals to 1. Moreover, the developed chain also evidently boosts the pre-processing stage of up to 3.78 times compared to the traditional CPU-based implementation.

ACKNOWLEDGEMENT

The authors would like to thank Satellite Technology Center, Indonesian National Institute of Aeronautics and Space (LAPAN) for supporting this work and providing the necessary data for testing purposes.

REFERENCES

[1] P. Escribano, T. Schmid, S. Chabrillat, E. Rodríguez-Caballero, and M. García, “Optical remote sensing for soil mapping and monitoring,” Elsevier Inc., 2017.

[2] M. Teke, H. S. Deveci, O. Haliloglu, S. Z. Gurbuz, and U. Sakarya, “A short survey of hyperspectral remote sensing applications in agriculture,” RAST 2013 - Proc. 6th Int. Conf. Recent Adv. Sp. Technol., pp. 171–176, 2013, doi: 10.1109/RAST.2013.6581194.

[3] C. Huang, Y. Chen, S. Zhang, and J. Wu, “Detecting, extracting, and monitoring surface water from space using optical sensors: a review,” *Rev. Geophys.*, vol. 56, no. 2, pp. 333–360, 2018, doi: 10.1029/2018RG000598.

[4] A. Wahyudiono and R. Madina. “Utilization of digital camera payload on LAPAN-A2 satellite for monitoring of toll road construction in Java Island,” Proceeding of SIPTEKGAN, vol. 20, pp. 242-249, 2017.

[5] W. Hasbi, Kamirul, M. Mukhayadi, and U. Renner, “The impact of space-based AIS antenna orientation on in-orbit AIS detection performance,” Appl. Sci., vol. 9, no. 16, 2019, doi: 10.3390/app9163319.

[6] W. Hasbi, “LAPAN-A2 (IO-86) satellite roles in natural disaster in Indonesia,” 70th International Astronautical Congress (IAC),” October 2019, pp. 3–8, 2020.

[7] A. Yusuf and S. Alawneh, “A survey of GPU implementations for hyperspectral image classification in remote sensing,” Can. J. Remote Sens., vol. 44, no. 5, pp. 532–550, 2018, doi: 10.1080/07038992.2018.1559725.

[8] D. Granata, A. Palombo, F. Santini, and U. Amato, “Noise removal from remote sensed images by non local means with OpenCL

algorithm," Remote Sens., vol. 12, no. 3, pp. 1–19, 2020, doi: 10.3390/rs12030414.

[9] K. P. Tejaswi, D. S. Rao, T. Nair, and P. A.V.V, "GPU accelerated automated feature extraction from satellite images," Int. J. Distrib. Parallel Syst., vol. 4, no. 2, pp. 1–15, 2013, doi: 10.5121/ijdps.2013.4201.

[10] J. Senthilnath, S. Sindhu, and S. N. Omkar, "GPU-based normalized cuts for road extraction using satellite imagery," J. Earth Syst. Sci., vol. 123, no. 8, pp. 1759–1769, 2014, doi: 10.1007/s12040-014-0513-1.

[11] Kamirul, P. R. Hakim, and S. Salaswati, "Statistica1-based Stripe Noise Correction on LAPAN Microsatellite Imagery," ICARES 2018 - Proc. 2018 IEEE Int. Conf. Aerosp. Electron. Remote Sens. Technol., vol. 5, pp. 72–77, 2018, doi: 10.1109/ICARES.2018.8547079.

[12] Kamirul, P. R. Hakim, and R. Madina, "Line-drop error correction for LAPAN-A2 microsatellite imagery," Proc. 2019 IEEE Int. Conf. Aerosp. Electron. Remote Sens. Technol. ICARES 2019, pp. 1–7, 2019, doi: 10.1109/ICARES.2019.8914346.

[13] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," Transactions on Image Processing, vol. 13, no. 4, pp. 600–613, 2017, doi: 10.1109/TIP.2003.819861.