On-board Implementation of Fractal Compression of Satellite Images using Distributed Networked GPUs

Munesh Singh Chauhan
Research Scholar
Pacific University
India

Sharmi S
Research Scholar
MS University
India

Abeer Marhoon Al-Sideiri
Research Scholar
Sultan Qaboos University
Oman

ABSTRACT

On-board image compression has been a growing trend in most recent satellite missions. Since majority of satellite applications deal with imagery; compression of images due to limited on-board data storage mediums has become a necessity. The idea of treating satellite imageries as fractals and then encoding them provides an efficient way of conserving bandwidth and per-bit storage costs. Fractal encoding is characterized by slow encoding times which somehow had hindered its popularity in spite of its impressive compression ratio scaling many orders as compared to JPEG. In order to circumvent this handicap, Fractal compression is implemented using powerful GPUs (Graphical Processor Units) that are capable of reaching astronomical computing speeds of around 900 GFLOPS (Quadro Graphic Cards from Nvidia^{TM}) with internal memory bandwidth ranging to 100 GB/s. This astounding parallel capability is probed to be used on board systems, providing much needed boost for image compression. As the decoding part of compressed fractal images is almost instantaneous, this part can be handled without any specific hardware at the ground station level. Further, the issue of on-board data storage mechanisms is discussed with emphasis on use of HDD instead of SSD and flash memories. In sum, the prime aim is to provide a seamless image compression mechanism coupled with de-compression at ground station level thus providing real-time streaming of satellite images from satellite to the ground.

General Terms

Fractal Image Compression, Graphical Processing Unit (GPU), Satellite images, On-board systems, et. al.

Keywords

Fractal images, CUDA – Compute Unified Device Architecture, Discrete Cosine Transform (DCT), RMS (root mean square value), change detection, panchromatic images, and multi-spectral images.

1. INTRODUCTION

Satellites play an important role in weather forecasting by providing high altitude view of the events unfolding on the earth's surface in the form of remotely sensed images on different bands [1]. These images can selectively be used for analyzing specific climatic or geographical details based on various classifications. In order to provide information that is real-time or near-real-time satellites need augmented on-board computing facilities that can process and disseminate imageries on-the-fly. This can be achieved using two critical enhancements; by providing multi-threaded compute intensive hardware in the form of GPUs (Graphical Processing Units) fitted on-board and secondly providing fast memory transfer data buses coupled with fast storage facility. As the image data consists of pixels that can be individually manipulated, GPUs provide a perfect environment due to the presence of many-cores that can process many threads in parallel.

Almost the entire remotely-sensed images have high percentage of self-similar components thus being suitable candidates for fractal representation and compression. By combining the fractal compression of remotely-sensed images with the "on-board" GPU processing facility; improved bandwidth utilization along with provision for high-fidelity image transfer to earth stations can be provided.

2. GPU COMPUTING ARCHITECTURE

GPU computing framework involves porting compute-intensive parts of a program to the GPU which stands for Graphical Processing Units. GPUs over the years have evolved into massively-parallel multicore systems that are used to solve compute-intensive real-life problems such as bio-engineering, real-time rendering to exploring new life forms on far away galaxies [2,8]. GPU in combination with the CPU also termed as GPU+CPU help in executing compute-intensive codes in parallel whereas the sequential part of the code continues it execution in the CPU. The GPUs are built on a fundamental proven concept that parallelism can be achieved more efficiently using simple, smaller cores that consume less power and provide multi-dimensional parallel capability. Initially, GPUs were extensively used for image processing and rendering but later it was realized that their application extended far beyond it.

Nvidia^{TM} has provided CUDA (Compute Unified Device Architecture) framework that can harness the GPUs capability in varied computing systems. The GPU representation can be described using the following two models:

1. Programming Model and the

2. Memory model

The scalable programming model provides for the parallel execution of certain portions of an application on a device called *kernels*. The *kernels* can be executed by arrays of CUDA threads. The threads are organized as grids of blocks that can be executed in parallel.

Now coming to the memory model, CUDA arranges runtime memory (as depicted in Figure 1), in the form of registers, local memory, shared memory, global memory and host

(CPU) memory. Each thread has its own registers and local memory. All threads in a block can access shared memory.
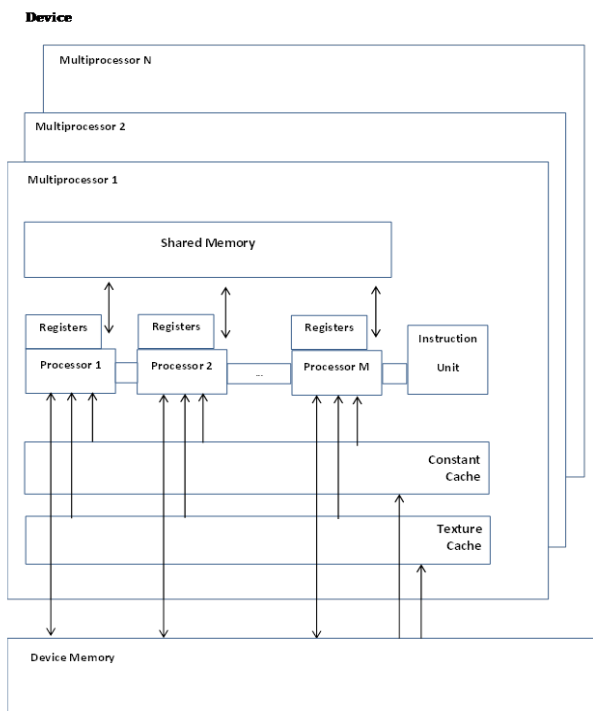
Fig 1: CUDA Memory Model

3. REVIEW OF PREVALENT IMAGE COMPRESSION TECHNIQUES

Image compression can be divided into lossy and lossless categories. As obvious from the term lossless, the encoded image when decompressed exactly resembles the original image. Lossless image compression is least efficient in terms of storage whereas lossy compression techniques exploit redundancies in a given image. Based on redundancies, following classifications of lossy compression techniques are possible:

1. **Spatial redundancy**: Spatial correlation methods such as predictions and transformations are employed to realize spatial redundancies. An example of prediction-based technique is DPCM (Differential Pulse Code Modulation) whereas that of transformations is DCT (Discrete Cosine Transform) and DWT.

2. **Statistical redundancy**: For example Huffman coding and arithmetic coding.

3. **Human vision redundancy**: This involves removing high-frequency components as these are indiscernible to human eye.

In general most of the popular image compression techniques use transforms based encoding such as DCT and DWT. The rest of the compression techniques are prediction-based, such as DPCM, LPEG-LS, lossless JPEG and Rice-based systems.

The above discussion is very succinct and gives a brief overview of the present-day contemporary, image compression algorithms. Exhaustive analysis and the use of these compression algorithms can be found in the work of Guoxia Yu, et. al.[3]. Till today, no fractal compression has been attempted on any launched platforms. The relevance of treating remote images as fractals and subsequent encoding can revolutionize the bandwidth capacity and cater to greater transmitted image volume over the present hardware. The brief review of fractal image compression in the next section with suitable modifications for on-board processing is proposed.

4. FRACTAL IMAGE COMPRESSION ON-BOARD SYSTEMS

Fractal image compression for long has been actively researched for compresion of stored satellite image data and then subsequent distribution [4]. The prime motivation for fractally compressing satellite image data is the high compression rates along with better image fidelity. Since the perception of image quality is quite subjective but this factor cannot be discarded as the satellite data is interpreted in different ways for different applications. Due to the very low root mean square (RMS) values obtained from comparing the compressed fractal image with the original, it provides a competitive and commercial potential for the inclusion of fractal compression in high-fidelity applications. It has been observed in fractal compression that signal to noise ratio (SNR) is comparatively better for smaller image sizes (such as 256×256 and 512×512), but the advantage lies in the fact that compression achieved with the given SNR, scales uniformly with the image size. This aspect is not so observed in the case of other commercially available image compression softwares that are mostly based on discrete cosine transforms. Thus, we can expect widespread potential for use of fractal compression in data-intensive imageries, such as obtained from a satellite.

Since fractal compression in general takes more time due to the search-and-compare overhead of the partitioned components of an image, it is proposed that the real-time processing of fractal images that are basically sensed arrays of CCD sensors installed on the EO (Earth Observing) satellites be conducted on-board using GPUs. In the current researches, the idea of intelligent EO satellites is debated and being highly explored. Now with the presence of satellites with a resolution from 1m to 15m in panchromatic, multispectral and radar format, the satellites are envisioned to be treated similar to a TV relay stations where the ground user may tune to any of its channels thus utilizing multi-functional systems on-
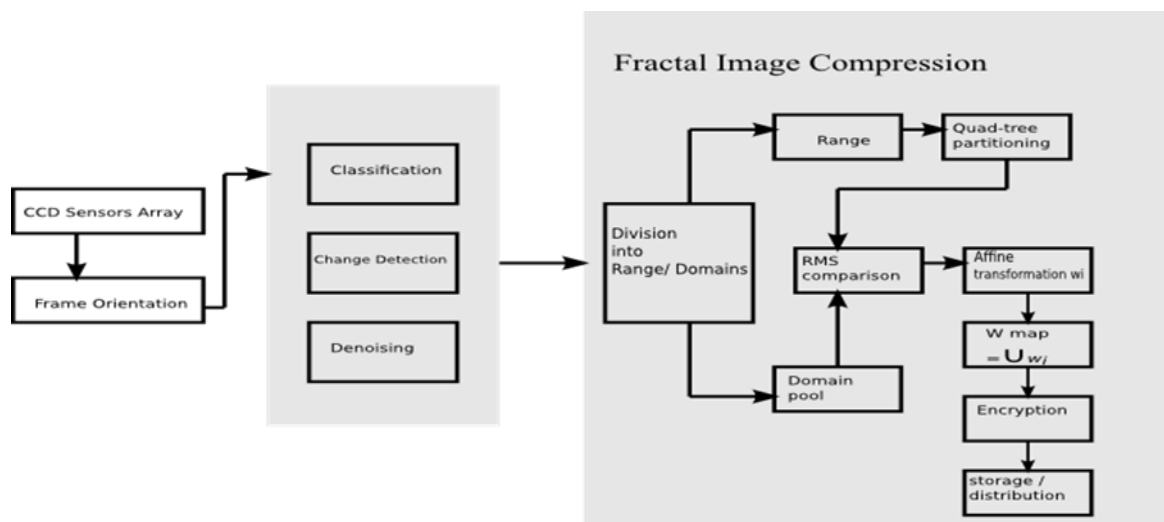
Fig 2: Block Diagram of proposed on-board Fractal Image Compression

board satellites [1]. The focus has shifted from data gathering to information generation. The proposed space-based architectural for dynamic fractal on-board compression is shown in Figure 2.

The on-board data capability will consider components similar to FIEOS [1] with specialized hardware for fractal processing. These systems will include: (a) Parallel Fractal Processor (GPUs), (b) Data Management Processor, (c) Data Transmission System, (d) Resource controller, (e) Communication System, and (f) platform mounted with sensors.

On-board processing of fractal image starts with the pre-processing stage (Figure 2) such that subsequent fractal compression may be implemented in an efficient manner. These steps consist of *image classification*, *change detection*, and *denoising*. Image classification and change detection finds usage in disaster management. Since satellite images are required for various applications, image classification helps in segregating images on the basis of their utility. Besides sometimes an application may require an image in raw format thus bypassing any compression & pre-processing. The prime aim of a raw-image requirement is based on the assumption that certain applications require processing of images entirely at the ground station level. Another very crucial step in pre-processing stage is termed as *change detection*. In other words the data will not be transmitted unless there has been a change in the image over a period of time during subsequent satellite pass. This approach frees up bandwidth substantially. This factor complements fractal compression as image is partitioned into different sizes (or tiles) depending upon the partition algorithm employed. Considering this view, the on-board fractal compression can be broadly categorized into two important domains:

1. Only those partitions that have significant changes (as compared to their previous image) will be compressed. Hence from this analogy, image data that is compressed fractally,

need not be assembled in one lot and then subsequently sent to the GDPC (Ground Data Processing Centre). The data can then be transmitted in parts in real-time with only those image segments following, that have significant changes. The final compression task can be allocated to the GPUs at GDPC. This concept is a topic under active research in our next paper.

2. The second part is the core of this paper that harnesses the parallel capability of the on-board GPUs and provides real-time fractal compression on the fly thus mitigating the concerns of bandwidth utilization typical of large volume image data.

Fractal compression can be implemented in software, hardware or both. It depends on the satellite development budget and availability. Many on-board compression algorithms have been developed partly in software (DSP) and the rest in hardware (FPGA). The hardware implementation definitely boosts efficiency but re-configurability is limited [5]. The proposed fractal compression envisages use of software implementation using GPU hardware as shown in Figure 1 thus providing maximum scope for re-configurability.

The FIC Block (Figure 2) shows fractal compression process of PAN (Pan-chromatic) and MS (Multi-spectral) images. The images are processed serially tile-by-tile. The main aim is to ensure that the FIC block is re-configurable in order to cater to new improvements in the fractal compression algorithms and related processes.

The processed data may be stored on-board mass storage devices (such Hard Disk Drives- HDD) or sent to ground. In some cases as pointed out before the data need to be stored in a raw state and thus there is no need to compress it. All these issues are handled at the level of pre-processing stage.

The Experiment: An analysis was conducted on two sets of image data: PAN and MS images drawn from satellites ALOS and SPOT6. The PAN image experimental data set (with and

without radiometric calibration [6]) was obtained from ALOS satellite having resolution of 2.5 meter whereas MS image data-set (with and without radiometric calibration) came from SPOT6 satellite with a resolution of 6 meters. The results are outlined in Table 1 and Table 2 for PAN and MS fractal image compression respectively.

Compression ratio	Fractal encoding-Quadtree		Fractal encoding-HV partitioning	
	PAN	PAN-R	PAN	PAN-R
al00234tp	2.567	2.634	3.109	3.120
al00013tp	2.459	2.781	3.319	3.223
al00569tp	3.119	3.450	2.978	2.658
al00743tp	3.072	3.566	4.013	3.896
al00113tp	3.884	3.901	3.672	3.419
AVE	3.020	3.266	3.418	3.263

Table 1: Fractal compression ratio of PAN images

(PAN → without radiometric calibration, PAN-R → with radiometric calibration)

Compression ratio	Fractal encoding-Quadtree	Fractal encoding-HV partitioning
Sp600144	2.89	2.77
Sp600245	2.18	1.87
Sp600498	3.17	3.09
Sp600081	3.73	3.66
Sp600564	3.84	3.57
AVE	3.16	2.99

Table 2: Fractal compression ratio of MS images

Results: From Tables 1 and 2 it is observed that fractal compression rates using GPUs is at par with JPEG compression rates if not better. Besides, the compression rates of both PAN and MS images are similar. In spite of having same compression rates as the conventional JPEG, fractal compression provides seamless scaling of decoded images with no loss in detail. This feature is completely absent in other compression algorithms. Since the satellite images provide vital information in all domains (geology, hydrology, vegetation, soil moisture, cloud formation, etc) seamless scaling without loss in image quality can immensely benefit applications that look for micro-details [7] in satellite images.

Further, since the onboard compression takes place in a hostile space environment the error correction has to be an integral part of any compression technique. Martin Sweeting in his work [8] states about having uncompressed reference samples between sets of data. Thus the error is limited to a certain region called ICR (*Independent Compression Region*) [9]. The proposed fractal compression will use the modified version of ICR where any error that occurs will result in re-computation of domain-range match of the image partitions sent prior to the occurrence of error. This will lower the overhead of re-computation of the entire partitions. This error handling can be employed in real-time image streaming of data from onboard link to ground using GPUs [10].

5. CONCLUSION

On-board fractal compression of satellite images has immense potential benefits that are yet to be realized. Though fractal compression ratios are closer to the conventional compression algorithms if not better but due to its undiminished fidelity of fractal images when scaled, it opens avenues for detailed views of images not seen so far. One of the possible applications of fractal compression lies in the domain of real-time, true color moving images beamed from satellites with high spatial resolution for 3G and GPS services.

6. REFERENCES

[1] Zhou G., Kafatos M., Future Intelligent Earth Observing Satellites, *Proc. SPIE* 5151, Earth Observing Systems VIII, 1 (November 13, 2003)

[2] Si X., Zheng H., High Performance Remote Sensing Image Processing Using CUDA, 3rd International Symposium on Electronic Commerce and Security, 29-31 July, 2010

[3] Yu G., Vladimirova T., Sweeting M. N., Image Compression Systems On Board Satellites, http://dx.doi.org/10.1016/j.actaastro.2008.12.006

[4] Fisher Y., Bielefeld B., Lawrence A., Greenwood D., NETROLOGIC Inc., Fractal Image Compression, Contract # N00014-91-C-0117, Quaterly Progress Report 8-12-9

[5] Antonio Lopes F., Roberto d'Amore, A low complexity Image Compression Solution for Onboard Space Applications, SBCCI '10 ACM Proceedings of the 23rd symposium on Integrated circuits and system design

[6] Guoxia Yu, Tanya Vladimirova, Martin Sweeting, An Efficient On-Board Lossless Compression Design for Remote Sensing Image Data, Geoscience and Remote Sensing Symposium, 2008. IGARSS 2008. IEEE International

[7] C.Lambert-Nebout, C.Latry, G.A.Moury, C.Parisot, M. Antonini, M.Barlaud, On-board optical image compression for future high-resolution remote sensing systems, in: Proceedings of SPIE on Applications of Digital Image Processing XXIII, vol.4115, December2000, pp.332–346.

[8] G. Yu, T. Vladimirova, X. Wu, M.N. Sweeting, A new automatic on-board multispectral image compression system for Leo Earth observation satellites, in: The 15th IEEE International Conference on Digital Signal Processing, 2007, pp. 395-398

[9] Kiely A., Klimesh M., The ICER Progressive Wavelet Image Compressor, IPN Progress Report 42-155, November 15, 2003

[10] NVIDIA, CUDA C Programming Guide, PG-02829-001_v5.0, October 2012, Design Guide