International Journal of Computer Applications Technology and Research
Volume 2– Issue 4, 424 - 430, 2013, ISSN: 2319–8656

High Performance Computing for Satellite Image Processing and Analyzing – A Review

Mamta Bhojne
Advanced Computing Training School (ACTS),
Centre for Development of Advanced Computing
(C-DAC), Pune, India.

Abhishek Chakravarti
Advanced Computing Training School (ACTS),
Centre for Development of Advanced Computing
(C-DAC), Pune, India.

Anshu Pallav
Advanced Computing Training School (ACTS),
Centre for Development of Advanced Computing
(C-DAC), Pune, India.

Sivakumar V
Geomatics Solutions Development Group,
Centre for Development of Advanced Computing
(C-DAC), Pune, India.

Abstract: High Performance Computing (HPC) is the recently developed technology in the field of computer science, which evolved due to meet increasing demands for processing speed and analysing/processing huge size of data sets. HPC brings together several technologies such as computer architecture, algorithm, programs and system software under one canopy to solve/handle advanced complex problems quickly and effectively. It is a crucial element today to gather and process large amount of satellite (remote sensing) data which is the need of an hour. In this paper, we review recent development in HPC technology (Parallel, Distributed and Cluster Computing) for satellite data processing and analysing. We attempt to discuss the fundamentals of High Performance Computing (HPC) for satellite data processing and analysing, in a way which is easy to understand without much previous background. We sketch the various HPC approach such as Parallel, Distributed & Cluster Computing and subsequent satellite data processing & analysing methods like geo-referencing, image mosaicking, image classification, image fusion and Morphological/neural approach for hyper-spectral satellite data. Collective, these works deliver a snapshot, tables and algorithms of the recent developments in those sectors and offer a thoughtful perspective of the potential and promising challenges of satellite data processing and analysing using HPC paradigms.

Keywords: Satellite Image, Remote Sensing, High Performance Computing, Parallel Computing, Distributed Computing & Cluster Computing

1. INTRODUCTION

High Performance Computing (HPC) is the recently developed technology in the field of computer science, which evolved due to meet increasing demands for processing speed. HPC brings together several technologies such as computer architecture, algorithm, programs and system software under one canopy to solve advanced complex problems quickly and effectively. This technology focuses on developing and implementing methods like parallel processing, cluster processing and distributed processing for solving problems. Parallel processing is a computing approach to increase the rate at which the set of data is processed by processing different parts of the data at the same time [1]. Unlike other methods where data in inputted in memory system in a step by step manner, distributed processing uses parallel processing on multiple machines, where data is distributed to all parts of the memory system at once. Parallel computing may be seen as a particular tightly coupled form of distributed computing [2] and distributed computing may be seen as a loosely coupled form of parallel computing [3]. In cluster computing many CPUs hooked up via high speed internet connections to a central server which gives each of them several task [1]. With the advancement of satellite remote sensing technology we are getting high spatial, spectral and radiometric resolution images with a huge data available. But problems occur when Remote Sensing image processing speed falls far behind which means that abundant data cannot be translated into useful information in time. Recently, application of HPC technology is getting more importance in remote sensing research work. The utilization of HPC systems in remote sensing applications has become more and more widespread in recent years [4]. HPC is able to improve the computing speed to a great extent in massive data processing, which makes itself an effective way to solve the problem of processing efficiency in remote sensing data. In this paper we present various techniques and methods of High Performance Computing for remotely sensed satellite image processing and analyzing. The following sections briefly describe the High Performance Computing technology for remote sensing data processing and analyzing methods.

2. PARALLEL COMPUTING

Parallel processing is the simultaneous processing of the same task on two or more microprocessors in order to obtain faster results. The computer resources can include a single computer with multiple processors, or a number of computers connected by a network, or a combination of both. The processors access data through shared memory. With the help of parallel processing, a number of computations can be performed at once, bringing down the time required to complete a project. Parallel processing is particularly useful in projects that require complex computations [5]. Han S.H. *et al.* (2009) explained that parallel processing system denotes a multiple-processor computer system consisting of centralized multiprocessors or multi-computers [6]. Figure-1 shows task based parallel processing workflow of automatic geometric correction (step 0), image matching (step 1) and Digital Surface Model generation (step n) using various data / block.

In parallel computing more than one processor is required to perform any task. There are two basic types of parallel computer systems *i.e.*, shared memory multi-computers (SMMC) and message passing multi-computers (MPMC) [7].

The difference between these two types is based on their memory storage unit. In SMMC memories are shared among computers, which means multi-computers share a uniformly coded storage unit and data exchange is realized by addressing operations. Whereas, MPMC uses network to connect computers or processors and each computer has its own storage unit which cannot be accessed by other computers [8].

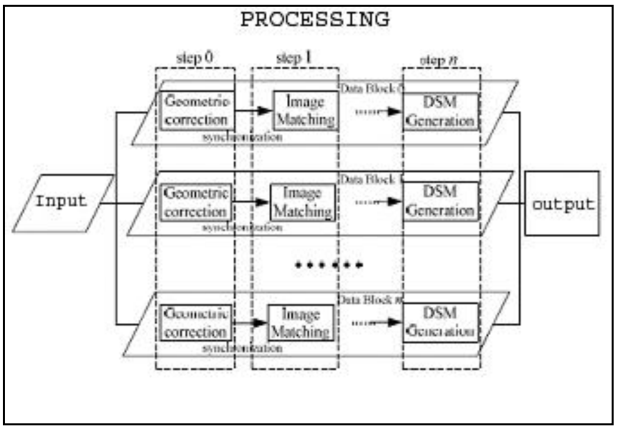

Figure 1. Task based parallel processing workflow (modified after, Hangye Liu *et al.*, 2009).

2.1 Image fusion

Image fusion is the process of merging multi-spectral image having high spectral resolution and pan-chromatic image having high spatial resolution (co-georeferenced). There are different algorithms and operations used for image fusion like Brovey, SFIM, HFM, Wavelet, HIS, Gram-Schmidt, PCA, etc. Yang J. H., *et al.*, (2010) explained that the parallel processing framework can be applied to most image fusion algorithms, which are divided into three categories such as component substitution (CS), modulation based fusion techniques and multi resolution analysis (MRA) based fusion techniques [9]. On satellite image fusion, algorithms can be examined in main four steps i.e., i) Co-register of MS and Pan image, ii) Upscale (interpolation) of MS image, iii) Gather spatial image in Pan image and iv) Merge spatial details with MS image [10]. They analyzed fourteen data fusion methods, execute both serial and parallel algorithms and compare execution times and quality performances. After experimenting serial and parallel algorithm they concluded that all parallel algorithms performed on average 4.4 times faster than serial algorithms with minimum 1.75 and 4.4 times faster. Yang Jinghui *et al.*, (2012) proposed that the parallel processing mechanism can divide an entire image into different blocks which are dispatched to different processing units [11]. Thus the processing efficiency is improved. Although, splitting the computation to more processing threads shortens the executing time but, it also increases the additional cost caused by inefficient memory usage when number of threads increases. In order to check the efficiency of different fusion methods, Alper *et al.*, (2013) applied indexes like Spectral Angle Mapper (SAM), Root Mean Square Error(RMSE), Relative Average Spectral (RASE) and Erreur Relative Globale Adimensionelle de Syntheses (ERGAS) [10]. SAM is used to measure the angle between the spectral vector of fused MS bands and original MS bands to analyze the spectral similarity [12]. Smaller results express high similarity between the images and higher express low similarity [13].

$$SAM(i) = arccos\left(\frac{\langle \hat{v}_i, v_i\rangle}{\|\hat{v}_i\|\|v_i\|}\right) \tag{1}$$

v^= spectral vector of fused image band

I = band

RMSE defines an error between the reference image and fused image for each band. The lower value or RMSE shows the higher spectral quality.

$$RMSE(i) = \frac{1}{tP}\sqrt{\sum_{k=1}^{tP}\left(M_i(k) - \hat{M}_i(k)\right)} \tag{2}$$

M = spectral vector of referenced image

tP =total pixels

RASE expressed the average implementation of RMSE of each spectral band,

$$RASE = 100\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{RMSE^2(i)}{\mu_i}\right)} \tag{3}$$

μi= mean radiance of n spectral bands of the reference image

Lower values for ERGAS represent higher spectral quality,

$$ERGAS = 100\frac{h}{l}\sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\frac{RMSE^2(i)}{\mu_i}\right)} \tag{4}$$

h= resolution of pan image

l= resolution of MS image

They further explained that the variation in performance of each test differs according to characteristics of methods and their algorithms, hardware limits, cache memory usage, hyper threading etc. They concluded that the best result is observed on Gram-Schmidt followed by IHS- Wavelet hybrid method.

2.2 Image classification

Image classification is the most important part of digital image processing. The intent of the classification process is to categorize all pixels in a digital image into one of several land cover classes or themes. This categorized data may then be used to produce thematic maps of the land cover present in an image [14]. There are two types of image classification- supervised and unsupervised. Supervised classification makes use of the training samples. While in unsupervised classification natural clustering or grouping of the pixel values i.e., gray levels of the pixels are observed. Smit M. *et al.*, (2000) described that technology to rapidly process imagery data into useful information products has not kept pace with the rapidly growing volume and complexity of imagery data increasing available from government and commercial sources. Significant processing speed improvements have been achieved by implementation of classification methods on the highly parallel integrated virtual environment (HIVE) - a Beowulf class system using parallel virtual machine software [15].

Kato Z. *et al*., (1999) dealt with the problem of unsupervised classification of images modeled by Markov Random Fields (MRF). They worked on parameter estimation methods related to monogrid and hierarchical MRF models using some iterative unsupervised parallel segmentation algorithms. They described algorithms which have been tested on image segmentation problems [16]. Also comparative tests have been tested on noisy synthetic data and on real satellite images. The algorithms were implemented on a Connection Machine CM200 [17, 18]. They compared the obtained parameters and segmentation results to the supervised results presented by Kato Z. *et al*., (1996) in the given Table 1 [19].

Table 1. Comparison of supervised and unsupervised classification (Number of misclassifed pixels)

Model	Image	Supervised	Unsupervised
Monogrid	Checkerboard	260 (1.59%)	213 (1.41%)
	Triangle	112 (0.68%)	103 (0.63%)
Hierarchical	Checkerboard	115 (0.7%)	147 (0.9%)
	Triangle	104 (0.63%)	111 (0.68%)

The result shows that unsupervised algorithms provide results comparable to those obtained by supervised segmentations, but they require much more computing time due to hyper parameter estimation and they are slightly more sensitive to noise. The main advantage is that unsupervised methods are completely data-driven where only input parameter is the number of classes.

2.3 Image mosaicking

Hongyu Wang (2005) has explained that image mosaicking is the process of combining a set of small images into a larger composite image [20]. However, it is very complex to mosaic multiple small images because individual images must be projected into a common coordinate space, overlap between images has to be calculated, the images should be processed so that the backgrounds match, and images composed while using a variety of techniques to handle the presence of multiple pixels in the same output space. To accomplish these tasks, a suite of software tools called Montage has been developed. The modules in this suite can be run on a single processor computer using a simple shell script, and can additionally be run using a combination of parallel approaches. These include running MPI versions of some modules, and using standard grid tools. In the latter case, processing workflows are automatically generated, and appropriate data sources are located and transferred to a variety of parallel processing environments for execution. As a result, it is now possible to generate large-scale mosaics on-demand in timescales that support iterative, scientific exploration [21]. Yan Ying Wang *et al*., (2010) described that image mosaic for large scale RS images, the registration and blending of mosaic is I/O sensitive and time consuming. They proposed an Optimized Image Mosaic Algorithm with Parallel I/O and Dynamic Grouped Parallel Strategy Based on Minimal Spanning Tree to solve the problems associated with image mosaicking. An effective parallel strategy of data splitting is adopted in the time-consuming part, registration and blending. In addition, the multi-thread parallel I/O strategy which is overlapping I/O and computing time is adopted to speed up the algorithm efficiency. Its outstanding parallel efficiency and perfect linear speedup is shown through experimental and comparative analysis [22].

2.4 Morphological /neural approach for hyperspectral satellite image processing

Valencia D. *et al*., (2007) demonstrated new parallel processing methodologies for hyper spectral image processing based on neural architectures and morphological concepts [23]. The computational performance of the proposed methods is demonstrated using real analysis scenarios based on the exploitation of AVIRIS data using two parallel computer systems and SGI Origin 2000 multicomputer located at the Barcelona Supercomputing Center (BSC) and the Thunderhead Beowulf cluster at NASA's Goddard Space Flight Center (NASA/GSFC). They developed a new parallel morphological /neural approach for hyper spectral image classification and specifically discuss implementation aspects using several commodity cluster-based architectures. They proposed methods for hyper spectral analysis which can be included in the category of spectral un-mixing and classification approaches respectively [24]. Valencia D. *et al*., (2007) described classification problem of spectral mixing, and then introduce a morphological operations to solve the problem using SOM (Self Organizing Map) and end member extraction-based approach/ algorithms [23]. Based on the morphological concept they proposed Automated End member Extraction Algorithm (AMEE) method, and allows soft classification of hyper spectral images in fully automated fashion. In addition to this they discussed parallelization strategies for AMEE and SOM algorithms. The proposed parallel algorithm fully exploits the underlying parallelism inherent in image processing methods which, minimizes the communication between processors [25]. Execution time (in seconds) of the AMEE algorithm at the SGI Origin 2000 multi-computer for several combinations of number of iterations. I_{MAX}, and number of processors, N is given in the Table 2 & 3.

Table 2. AMEE algorithm (time in seconds) in SGI origin 2000 (From Valencia D. *et al.*, 2007).

N	I_{MAX}=1	I_{MAX}=3	I_{MAX}=5	I_{MAX}=7
1	372	1066	1809	2476
2	182	522	864	1178
4	89	252	429	569
8	264	143	338	293

Execution time (in seconds) of the AMEE algorithm at the Thunderhead Beowulf cluster for several combinations of number of iterations. I_{MAX} and number of processors N. They concluded that parallel computing at the massively parallelism level, supported by message passing, provides a unique framework to accomplish the above goals. For this purpose, computing systems made up of arrays of commercial off-the-shelf computing hardware are a cost-effective way of exploiting this sort of parallelism in remote sensing applications. Specifically, the proposed MPI-based parallel implementation minimizes inter-processor communication overhead and can be ported to any type of distributed memory system.

Table 3. AMEE algorithm (time in seconds) in Thunderhead Beowulf cluster (From Valencia D. *et al*., 2007).

N	I_{MAX}=1	I_{MAX}=3	I_{MAX}=5	I_{MAX}=7
1	311	947	1528	1925
4	124	321	557	685
16	45	95	144	156
36	26	46	61	71
64	19	29	41	43
100	12	20	26	29
144	9	15	20	23
196	6	11	17	20
256	4	10	14	18

3. CLUSTER COMPUTING

A computer cluster is a group of interconnected CPU's which are employed to process large datasets. The interconnection can be of many different types including via LAN, ftp server, Bluetooth network, Wi-Fi, etc. Computer cluster emerged as a result of convergence of a number of computing trends including the availability of low cost microprocessors, high speed networks and software for high performance distributed computing [1]. Clusters are usually deployed to improve performance and availability over that of single computer, while typically being much more cost effective than single computers of comparable speed or availability [26]. A cluster computing system is a compromise between a massively parallel processing system and a distributed system [27]. The architecture of cluster computing is given in the Figure 2.

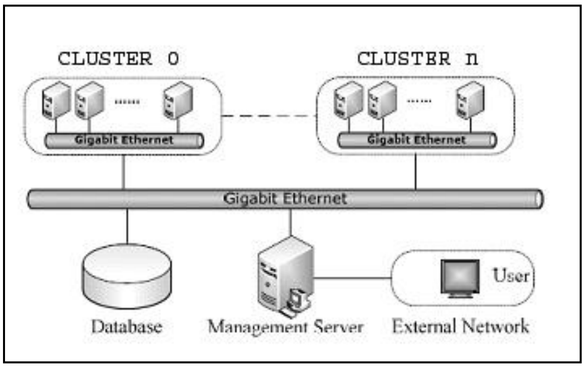

Figure 2. Architecture of remote sensing parallel processing system based on cluster computing (Modified after, Hangye Liu *et al*., 2009).

During recent years, cluster systems have played a more important role in the architecture design of high-performance computing area. Yuanli Shi *et al.*, (2012) stated that Satellite Environment Center, Ministry of Environment Protection of China has built a powerful cluster system which is designed to process massive remote sensing data of HJ-1 satellites automatically every day [28]. To verify the performance of cluster system, image registration has been chose to experiment with one scene of HJ-1 CCD sensor. The experiments of imagery registration show that it is an effective system to improve the efficiency of data processing, which could provide a response rapidly in applications. Wang Xuezhi *et al.*, (2010) have developed a web based data processing system based on Geospatial Data Abstraction Library (GDAL) which made the use of cluster computing and parallel computing [29]. The system achieved not only the online processing of 14 vegetation indices like NDVI and EVI, but also the online gap-fill algorithm for Landsat-7 SLC-off datasets.

Yang C.T. and Hung C.C. (2000) present the basic programming techniques by using PVM (Parallel Virtual Machine) to implement a message-passing program to utilize the parallelism of cluster of SMPs (Symmetric Multi Processor) [27]. The matrix multiplication of the parallel ray tracing problems is illustrated and the experiments are also demonstrated on Linux SMPs cluster. The program for matrix multiplication is given below:

```
for (i = 0; i < N; i ++ ) {                 /* can be parallelized */
    for (j = 0; j < M; j++ ){               /* can be parallelized */
    c[i] [j] = 0;
      for (k = 0; k < P; k++)
         c[i] [j] = c[i] [j] + a[i] [k] * b [k] [j];
   }
}
```

The matrix multiplication algorithm is implemented in PVM (Parallel Virtual Machine) using the master-slave paradigm. The experimental results showed that the highest speedup were 10.89 and 13.67 respectively for matrix multiplication of the PVMPOV (an unofficial version of Pov-ray), when the number of processors is 16, by creating 16 tasks on SMPs cluster. The results of this study will make theoretical and technical contributions to the design of a PVM program of a Linux SMP clusters for remote sensing data processing. It also shows that Linux/PVM cluster can achieve high speedups for applications.

4. DISTRIBUTED COMPUTING

Distributed computing is the process of aggregating the power of several computing entities to collaboratively run a single computational task in a transparent and coherent way, so that they appear as a single centralized system. Connecting users and resources in a transparent, open and scalable way is the main goal of distributed operating system [30]. Godfrey B.(2002) has described that distributed computing works by splitting the larger into smaller chunks which can be performed at the same time independently of each other [31]. The two main entities in distributing computing are the server and the many clients. A central computer, the server will generate work packages which are passed onto worker clients. The clients will perform the task, detailed in a work package data and when it has finished the completed work package will be passed back to the server. The working process of semi distributed scheduling policy is given in the Figure 3.

Processing image data generated by new remote sensing systems can severely tax the computational limits of the classic single processor systems that are normally available to the remote sensing practitioner. Operating on these large data sets with a single computer system, sometimes simplifying approximations are used that can limit the precision of the final result. Recent work at Pacific Northwest National Laboratory strongly suggest that a distributed network of inexpensive PCs can be designed that is optimal to deal with intensive computationally problems. The new type of distributed computing will remove computational constraints; image processing algorithms for remote sensed images are now being considered [32].

Geo referencing is basic function of remote sensing data processing. It is a process of assigning geographic information

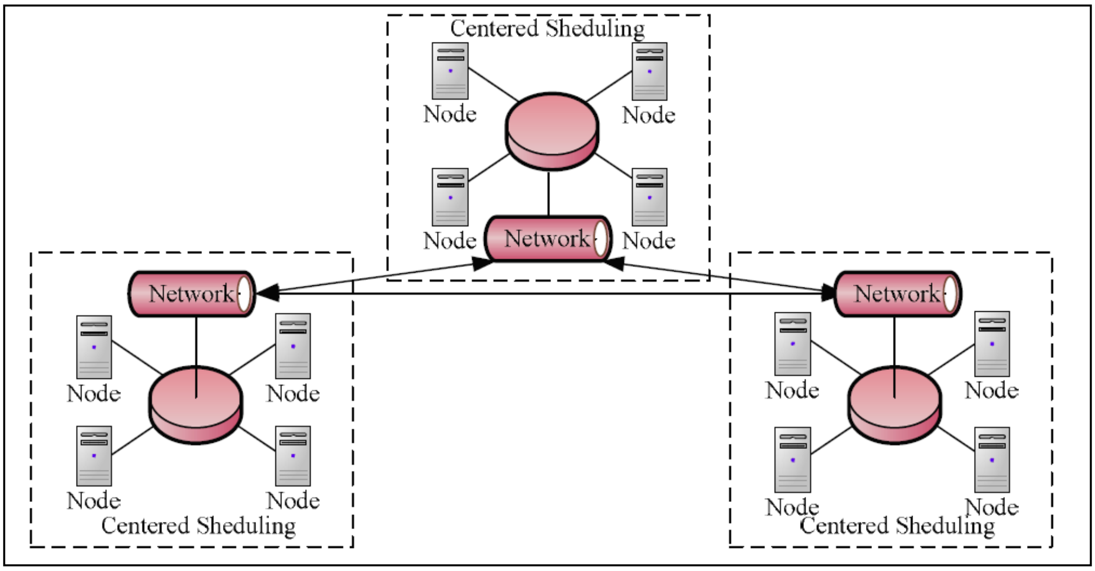

Figure 3. Semi distributed scheduling policy (modified after, Hangye Liu *et al*., 2009).

to an image. Knowing where an image is located in the world allows information about features contained in that image to be determined. This information includes location, size and distance. But it is very time consuming and computationally intensive process. To improve the efficiency of processing Yincui Hu *et al*., (2005) focuses on parallelization of remote sensing data on a grid platform [33]. As an important new field in the distributed computing arena, Grid computing focuses on intensive resource sharing, innovative applications, and, in some cases, high performance orientation. They performed their experiments on MODIS level 1B data. Two strategies were followed by them for geo referencing process viz. parallel rectification on grid and data partition strategy. They explained three components to rectify image, i.e. transformation model selection, coordinate transformation and resampling to correct every part of the large image. The partition strategy influences the process of efficiency and determines the merge strategy. According to feature of algorithm the applied backward decomposing techniques which comprises four steps i.e. partitioning the output array into equal sized block, computing geographical range of every block, finding GPCs triangulations contained with geographical range and extract block from original data in accordance with these triangulation. This extracted block is the data that will be distributed to producers. The experiment shows that data- parallel geo-reference is efficient especially for large-size data. The large data is decomposed into small parts and distributed to the Grid. The experiments indicate that Grid is efficient for data-parallel geo-reference.

Shamim Akhter *et al*., (2005) described a parallel approach in cluster computing and MPI (Message Passing Interface) parallel programming and provide results of experiments on studying the porting of remote sensing algorithm [34]. They used MPI as programming tool and all the codes were tested Beowulf cluster using GNU C compiler with the MPICH. implementation of MPI. They performed their experiment in compressed ASTER image and uncompressed MODIS image. The tasks are allocated to slave processor by a master processor. After getting data from input file, server put these data into 2D array which is distributed to different processors using either of two procedures, first, distributed each input pixel of a particular remote sensing image to corresponding processor and distributed row or column of that image at a time to the corresponding processor. Figure 4 shows flow of task allocation by master processor.

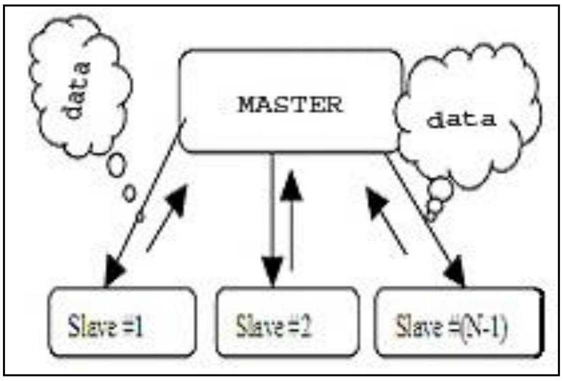

Figure 4. Client server paradigms (modified after, Shamim *et al*., 2005).

From these two experiments with a simple processing at two images size difference, it is observed that there is a point of convergence of all curves for a given image with an increasing number of operations applied.

5. CONCLUSIONS & FUTURE SCOPE

Methods discussed above are some of the techniques that are the subset of High Performance Computing (HPC) techniques that are being employed today to process large amounts of data. Thus different approaches can be employed for different projects. Parallel processing is particularly useful in projects that require complex computations. Parallel processing framework can be applied to most image fusion algorithms and hyper spectral image processing based on neural architectures and morphological model. Distributed computing is particularly useful when large amounts of data have to be processed within a given time period, keeping in mind the economic restrictions. A distributed network of inexpensive PCs can be designed that is optimal to deal with the type of computationally intensive problems encountered in processing remotely sensed images. Cluster computing is one of the most widely used HPC approach for processes such as geo-referencing, image transformation, image mosaicking, etc. Currently most of these approaches are limited to military and government organizations but private enterprises are also employing this technology at a rapid pace. Although, more research work requires on satellite data processing and analyzing over HPC platform for getting an enhanced and fast output for various remote sensing applications.

6. ACKNOWLEDGMENTS

The authors are thankful to ACTS, C-DAC, Pune for providing facilities and support to carry out this study. Authors would like to extend their gratitude to Ms. Asima Mishra, Joint Director–GSDG and Dr. Sunil Londhe GSDG, C-DAC, Pune for their encouragement.

7. REFERENCES

[1] WIKI. (2013). http://wiki.answers.com/Q, Accessed. June, 2013.

[2] Peleg D. (2000). Distributed Computing: A Locality-Sensitive Approach, SIAM, ISBN 0-89871-464-8:pp. 10. https://en.wikipedia.org/wiki/Distributed_computing. Accessed, June, 2013.

[3] Ghosh and Sukumar (2007). Distributed Systems – An Algorithmic Approach. Chapman & Hall/CRC. https://en.wikipedia.org/wiki/Distributed_computing. Accessed, June, 2013.

[4] Lee, C. A., Gasster, S. D., Plaza, A., Chang, C. I., & Huang, B. (2011). Recent developments in high performance computing for remote sensing- A review. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 4.3: 508-527.

[5] WISEGEEK. (2013). http://www.wisegeek.com/what-is-parallel-processing.html. Accessed, June, 2013.

[6] Han S.H, Joon Heo, Hong Gyoo Sohn and Kiyun Yu. (2009). Parallel Processing Method For A Airborne Laser Scanning Data Using a PC Cluster and a Virtual Grid. Sensors, 2009, 9: pp.2555-2573; DOI: 10.3390/s90402555. www.mdpi.com/1424-8220/9/4/2555/pdf. Accessed, June, 2013.

[7] Barry Wilkinson B. and Michael Allen (2002). Parallel Programming. China Machine Press. www.cse.ucsc.edu/classes/cmpe113/Fall02/slides1.4.ps. Accessed, June, 2013.

[8] Hangye Liu, Yonghong Fan, Xueqing Deng, Song Ji. (2009). Parallel Processing Architecture of Remote Sensed Image Processing System Based on Cluster. IEEE Image and Signal Processing 09. CISP '09. 2nd International Congress. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5300938. Accessed June, 2013.

[9] Yang J.H., J.X. Zhang, Li Haitao, Sun Yushan, Pu Pengxian.(2010). Pixel Level Fusion Methods for Remote Sensing Images: a Current Review,Technical Commission VII Symposium, Vienna, Austria. pp.680. http://www.isprs.org/proceedings/XXXVIII/part7/b/pdf/680_XXXVIII-part7B.pdf. Accessed, June, 2013.

[10] Alper A G., Adnan O, Meric Y., Sedef K P., Serdar B., and Mesut K. (2013). Remote Sensing Data Fusion Algorithms With Parallel Computing. . http://academia.edu/3266480/Remote_Sensing_Data_Fusion_Algorithms_with_Parallel_Computing. Accessed June, 2013.

[11] Yang Jinghui, Zhang Jixian (2012). A Parallel Implementation Framework For Remotely Sensed Image Fusion. ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences, Volume I-7, 2012 XXII ISPRS Congress, 25 August – 01 September 2012, Melbourne, Australia. http://www.isprs-ann-photogramm-remote-sens-spatial-inf-sci.net/I-7/329/2012/isprsannals-I-7-329-2012.pdf. Accessed, June, 2013.

[12] Yuhas R.H., Goetz A.F. And Boardman J.W. (1992). Discrimination Among Semi Arid Landscape End Members Using Spectral Angle Mapper (SAM) Algorithm. Summeries Of The Third Annual JPL Airborne Geosciences Workshop ,Vol. 1,Pasadena, CA:JPL Publication.pp.147-149. http://academia.edu/3266480/Remote_Sensing_Data_Fusion_Algorithms_with_Parallel_Computing. Accessed, June, 2013.

[13] Chikr M E Mezouar, N.Taleb, K.Kpalma and J. Ronsin (2011). An HIS Based Fusion For Color Distortion And Vegetation Enhancement In Ikonos Imagery. Geosciences And Remote Sensing, IEEE Transactions on,Vol.49,No.5. pp. 1590-1602. http://academia.edu/3266480/Remote_Sensing_Data_Fusion_Algorithms_with_Parallel_Computing. Accessed, June, 2013.

[14] SC.(2013). Remote sensing University Lecture note. http://www.sc.chula.ac.th/courseware/2309507/Lecture/remote18.html, Accessed, June, 2013.

[15] Smit M.,Garegnani J,Bechdol M and Chettri S.(2000). Parallel Image classification on HIVE. Applied imagery pattern recognition workshop,IEEE, 2000:39-46. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=953601&tag=1. Accessed, June, 2013.

[16] Kato Z., Zerubia J., Berthod M. (1999). Unsupervised parallel image classification using Markovian models. Pattern Recognition 32 (1999) 591Ð604, p.591-602. http://www.inf.u-szeged.hu/~kato/papers/pattrec99.pdf. Accessed, June, 2013.

[17] Hillis W.D. (1985). The Connection Machine. MIT press New York. http://www.inf.u-szeged.hu/~kato/papers/pattrec99.pdf. Accessed, June, 2013.

[18] TMC, Thinking Machines Corporation, Cambridge, Massachusetts, Connection Machine Technical Summary (1989). Version 5.1 ed. http://www.inf.u-szeged.hu/~kato/papers/pattrec99.pdf. Accessed, June, 2013.

[19] Kato Z, Berthod M, Zerubia J., A. (1996). hierarchical Markov random Field model and multi-temperature annealing for parallel image classification, Compute. Vision Graphics Image Process. Graphical Models Image Process. 58:18-37. http://www.inf.u-szeged.hu/~kato/papers/pattrec99.pdf. Accessed, June, 2013.

[20] Hongyu Wang (2005). Parallel Algorithms For Image And Video Mosaic Based Applications.p.1. http://athenaeum.libs.uga.edu/bitstream/handle/10724/8351/wang_hongyu_200508_ms.pdf?sequence=1. Accessed, June, 2013.

[21] Katz D.S., Nathaniel Anagnostou, G. Bruce Berriman, Ewa Deelmans, John Good Joseph C.Jaco, Arl Kesselman, Anastasia Laity, Thomas A. Prince, Gurmeet Singh, Mei Hui Su, Roy Williams (2006). Astronomical Image Mosaicking On a Grid: Initial Experiences. Engeneering the Grid Status and Perspective. Book - American Scientific Publishers. ISBN:1-58883-038-1. http://montage.ipac.caltech.edu/publications/montage_ETG.pdf. Accessed June, 2013.

[22] Yan Ying Wang, Yan Ma, Peng Liu, Dingsheng Liu and Jibo Xie. (2010). An Optimized Image Mosaic Algorithm with Parallel IO and Dynamic Grouped Parallel Strategy Based on Minimal Spanning Tree.

Proceeding Gcc ’10 proceeding of the 2010 ninth international conference on grid and cloud computing.pp. http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=05662698. Accessed, June, 2013.

[23] Valencia D., Plaza A., Martinez P., Plaza J. (2007). Parallel Processing of High Dimensional Images Using Cluster Computer Architectures. IJCA, vol.14, no. 1. pp 23-34. IJCA, Vol. 14, No. 1: pp.23-34. http://www.umbc.edu/rssipl/people/aplaza/Papers/Journals/2007.IJCA.Cluster.pdf. Accessed, June, 2013.

[24] Plaza A., Martínez P., Pérez R. and Plaza J. (2004). A New Approach for Mixed Pixel Classification in Hyperspectral Imagery Based on Extended Morphological Profiles. Pattern Recognition, 37:1097-1116. http://www.umbc.edu/rssipl/people/aplaza/Papers/Journals/2007.IJCA.Cluster.pdf

[25] Seinstra F.J., Koelma D., And Geusebroek J. M., (2002). A Software Architecture For Transparent Parallel Image Processing. Parallel Computing,28: pp.967-923. http://www.umbc.edu/rssipl/people/aplaza/Papers/Journals/2007.IJCA.Cluster.pdf. Accessed, June, 2013.

[26] Bader D., Pennington R. (2007). Cluster Computing Applications. The international journal of high performance computing. 15(2):181-185. http://en.wikipedia.org/wiki/Computer_cluster. Accessed, June, 2013.

[27] Yang C.T. and Hung C.C. (2000). Parallel Computing in Remote Sensing Data Processing. GIS DEVELOPMENT, AARS, ACRS, Image Processing. 1(1):1-6. http://www.a-a-r-s.org/acrs/proceeding/ACRS2000/Papers /OMP00-4.htm. Accessed June, 2013.

[28] Yuanli Shi, Wenming Shen, Wencheng Xiong, Zhuo Fu, Rulin Xiao (2012). High Performance Cluster System Design for Remote Sensing Data Processing. High-Performance Computing in Remote Sensing II. Proceedings of the SPIE, Volume 8539, article id. 85390N.

[29] Wang Xuezhi, Lin Qinghui, Zhou Yuanchun (2010). A Web-Based Data-Processing System For LandsatImagery. http://www.codata.org/10Conf/abstracts. Accessed, June, 2013.

[30] WORDIQ (2013). Distributed_computing. http://www.wordiq.com/definition/Distributed_computing, Accessed. June, 2013. Accessed, June, 2013.

[31] Godfrey B. (2002). Document, Primer on Distributed Computing. http://www.bacchae.co.uk/docs/dist.html. Accessed June, 2013.

[32] Petrie G. M., Dippold C., Fann G., Jones D., Jurrus E., Moon B. (2002). Distributed Computing Approach For Remote Sensing Data. (Conference) Symposium On Parallel And Distributed Computing, SPDP. http://academic.research.microsoft.com/Paper/2563627.aspx. Accessed, June, 2013.

[33] Yincui Hu, Yong Tang, Jiakui Tang, Shaobo Zhong and Guoyin Cai. (2005). Data Parallel Method for Georeferencing of MODIS Level 1B Data Using Grid Computing. 5th International Conference, Atlanta, GA, USA, May 22-25, 2005, Proceedings, Part III. http://rsgisforum.irsa.ac.cn/download/05_SCI/Data-parallel%20method%20for%20georeferencing%20of%20MODIS%20level%201B%20data%20using%20grid%20computing.pdf. Accessed, June, 2013.

[34] Shamim Akhter , Kiyoshi Honda , Yann Chemin , M. Ashraful Amin (2005). Experiments On Distributed Remote Sensing Data (Modis And Aster) Processing Using Optima Cluster. http://www.academia.edu/1760912/EXPERIMENTS_ON_DISTRIBUTED_REMOTE_SENSING_DATA_MODIS_AND_ASTER_PROCESSING_USING_OPTIMA_CLUSTER. Accessed, June, 2013.