Image Enhancement of Magnetic Resonance Imaging Under Clustering Environment

V. Murugan
Department of Computer Science
MSU Constituent College
Kadayanallur, Tirunelveli, India
smv.murugan@gmail.com

ABSTRACT

The process of image denoising aims at the removal of noise from an image. Noise in an image can be perceived as pixels with varying intensity values rather than original pixel values. The main reason for noise occurrence may happen during image acquisition or image transmission. Thus, practically it is not possible to escape from noise. The quality of an image has its impact over the degree of noise present in an image. Hence, it is necessary to remove or reduce such noise, in order to improve the quality of an image. This chapter is based on client-server architecture and the focus is rendered on salt and pepper and gaussian noise. The input image is partitioned into four parts by the server and all the partitions are shared with the client systems. The client systems process the partitioned input by denoising algorithms and submit the outcome to the server. The server unites all the denoised images together to arrive at a single denoised image. This work saves much time and the image is denoised in a matter of seconds.

Keywords-noise; filter; cluster; master-slave environment

I. INTRODUCTION

Image denoising is a challenging task as it is applicable to many branches of image processing such as image classification, image registration, and image segmentation and so on. Some of the main sources of noise are inadequate lightings, unfavourable sensor temperature; signal interference during image transmission etc., the quality of an image is inversely proportional to the presence of noise in an image. Thus, several algorithms have been proposed earlier for reducing or removing the noise present in an image. The major categories of noise are Salt and Pepper noise, Gaussian noise, speckle noise, uniform noise, shot noise and periodic noise [1].

The main objective of a denoising algorithm is to eliminate noise in an image in association with edge preservation. Initially, the noise present in an image is identified and then the corresponding denoising algorithm is applied to get rid of that noise.

If the image is affected by Salt and Pepper noise, then the entire image is degraded by black and white dots. The name ‘Salt and Pepper’ is rendered with respect to the noise resemblance. This type of noise may appear in an image because of the immediate changes in the signal or during image acquisition. The Salt and Pepper noise can also be called as Impulse noise, Spike and Random noise [2].

Gaussian noise is distributed all through the image and each pixel in the noisy image is the summation of original pixel value and the distributed Gaussian noise value. The process of image denoising is classified into two different techniques and they are spatial and transform domain filtering [3].

Spatial filtering is the simple concept for denoising and this is divided into linear and non-linear filters. Linear filters introduce smoothness into an image and it may miss the important detail of an image. Some of the linear filters are Wiener and mean filters. Non-linear filters eliminate the noise from an image. The best example for non-linear filter is median filter [4].

Median filter follows the concept of windowing. This filter eliminates the noise by calculating the median of the window and the center pixel of the image is modified with it.

A filtering technique is proposed in [5] for eliminating the Salt and Pepper noise in binary images is proposed. This technique works on the basis of computation over multi-diagonal binary matrix of noisy binary image and the thresholding operation is carried out.

In [6], an adaptive multi-column stacked sparse denoising auto-encoder is presented. Multiple stacked sparse denoising auto-encoders is combined by computing column weights and a network is trained to predict the optimal weights. The need of noise type detection is eliminated in this system and hence it is robust.

A neural network based region classification technique is proposed in [7]. The regions of an image is classified into homogeneous and texture regions. A neural network is trained based on the statistical parameters. The two classes namely the homogeneous and texture regions are denoised by shear lets and wavelets respectively.

A fuzzy based adaptive mean filtering is proposed in [8]. In this methodology, corrected and uncorrupted pixel is calculated by the membership value of all pixels. The value of the corrupted pixel is replaced by the mean of the uncorrupted pixels.

Motivated by the above mentioned works, it is planned to reduce the time taken to denoise an image and thus the client server architecture is employed. This system is effective in noise detection and removal.

II. PARALLEL COMPUTING AND DISTRIBUTED SYSTEMS

A. Parallel Computing

Parallel [9] is an alternative to solve problems that require large times of processing or handling amount of information in acceptable time. In the parallel processing program is able to create multiple tasks that work together to solve problem. The main idea is to divide the problem into simple tasks and solve them concurrently so that time is divided. Depending upon the requirement of the application and available budget, the selection of architecture is done. The parallelism can be applied in image processing applications by three main ways:

1) Data Parallel
2) Task Parallel
3) Pipeline Parallel

1. Data Parallel:

In data parallel approach [9], the data is divided and distributed among the computing units.The main challenge is efficient data decomposition and result composition. The main issue must be considered for efficient parallel execution is load balancing.Image data should be distributed among computing units in such a way that there will be less unnecessary communication among computing units and each unit gets approximately same load.

The data parallelism to image data can be applied using one of three basic ways:

i) Pixel parallel
ii) Row or column parallel
iii) Block parallel.

At present, the most of the parallel image processing applications use row/column parallel or block parallel

2. Task Parallel:

In task parallel approach [9], image processing instructions/ low level operations are groupedinto tasks and each task is assigned to a different computing unit. An image processing application consists of many different operations. The main challenge in task parallel approach is efficient data decomposition and result composition.

3. Pipeline Parallel:

If image processing application requires multiple images to be processed, then pipelineprocessing of images can be done. In pipeline processing, images will be in different stages at same time. A parallel program must have some features for a correct and efficient operation. These features include the following:

B. Distributed System:

There are two main architectures of distributed systems

1) The master slave
2) Peer to peer.

These are discussed below:

Master Slave:

The master slave architecture approach uses the “Distribute Compute and Gather” philosophy forparallel image processing. In this architecture approach, the master processing unit divides and distributes the image data to the slave processing units. All slave processing units work in parallel to achieve assigned task. Then, master processing unit gathers and assembles the image back.

Peer to Peer:

In peer to peer architecture, each participating entity has same capabilities and either entity caninitiate a communication. The participating entities make a portion of their resources directly available to other networked participating entities, without the need for central coordination.

III. PROPOSED ARCHITECTURE

In Image Denoising Client Server Architecture (IDCSA), client-server architecture is employed for sharing the workload with the resources available and to make use of all the resources properly. The input image is partitioned into four parts by the server and all the partitions are shared with the client systems. The client systems process the partitioned input by denoising algorithms and submit the outcome to the server. The server unites all the denoised images together to arrive at a single denoised image. This work saves much time and the image is denoised in a matter of seconds.

The system is depicted in Fig.1. The system is depicted more clearly in Fig. 2. The advantages of using this method are:

- We can use different denoising methods in each client system.
- Processing time is reduced.

The disadvantages of using this method are:

- If any of the clients is crashed, the total system fails.
- The server may get overloaded if more images are processed

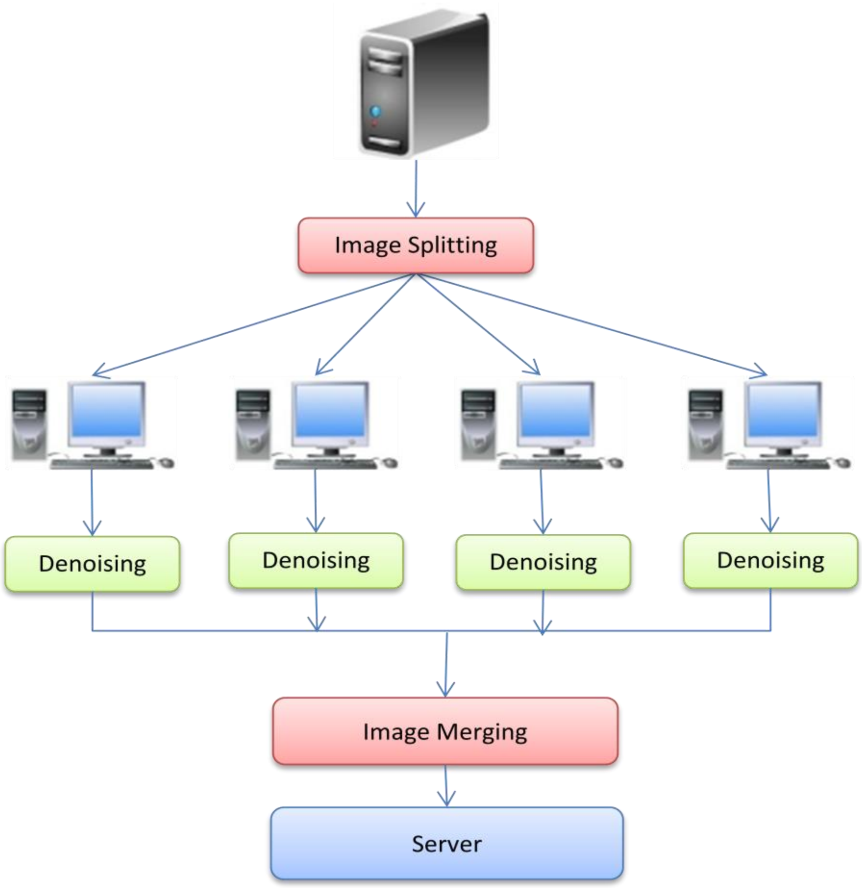

Figure 1: Overview of IDCSA

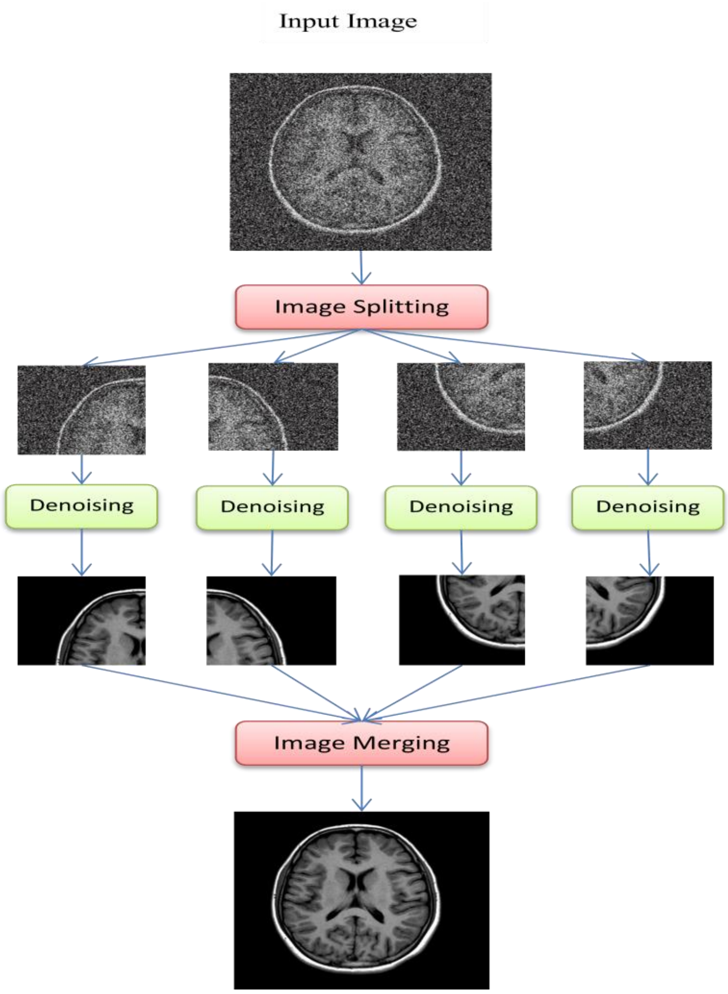

Figure 2: Image splitting and merging by IDCSA

IV. EXPERIMENTAL RESULTS AND PERFORMANCE ANALYSIS

The performance analysis, comparisons will be made by taking images with different image size. Different image quality metrics and other parallel computing parameters such as speedup, efficiency, serial time, parallel time, response time and resource utilization will be considered to evaluate the performance of IDCSA.

1. Speed up :

It is the ratio between sequential execution time and parallel execution time where the sequential time execution time is sum of total computation time of each task and parallel time execution is the scheduling length on limited number of processors.

2. Efficiency:

The efficiency of a parallel program is a measure of processor utilization. It is given as

$$EFF = S_p / N_p \tag{1}$$

Where,

S_p=Speedup,

Np=Number of processors.

3. Overheads:

Overheads of the parallel program can be measured as extra time needed for performing the computations

$$\text{Overheads} = \text{parallel time} - (\text{Serial time} / \text{No. of processors}) \tag{2}$$

4. **Fork Time:**

It is defined as time when data is distributed among the number of processors.

5. **Join Time:**

It is defined as time when result obtained from number of processors.

This section describes the results obtained with the parallel implementation IDCSA. This also deals with the environment used in the experiments, the test images, and results, along with the evaluation of the performance obtained with the parallelization.

Table 1 shows the performance comparison of client server system and single system. The values in the table are given in milliseconds.

Table 1 Performance analysis of single and client server system

System type	**Image Name**	**Image size**	**Fork time**	**Processing time**	**Join time**	**Total Processing Time**	**Efficiency**	**Speed up**	**Overhead**
Single system	MRI Image	512×512	5	8	3	16	86	71	63
Client server system	MRI Image	512×512	3	6	2	11	92	85	48

In the above Table 1, it is observed that the proposed architecture reduces the total processing time to 11ms for image of size 512 x 512. The overhead is also reduced by converting to client server system. Figure 3 shows the performance comparison chart of single system and client server system.

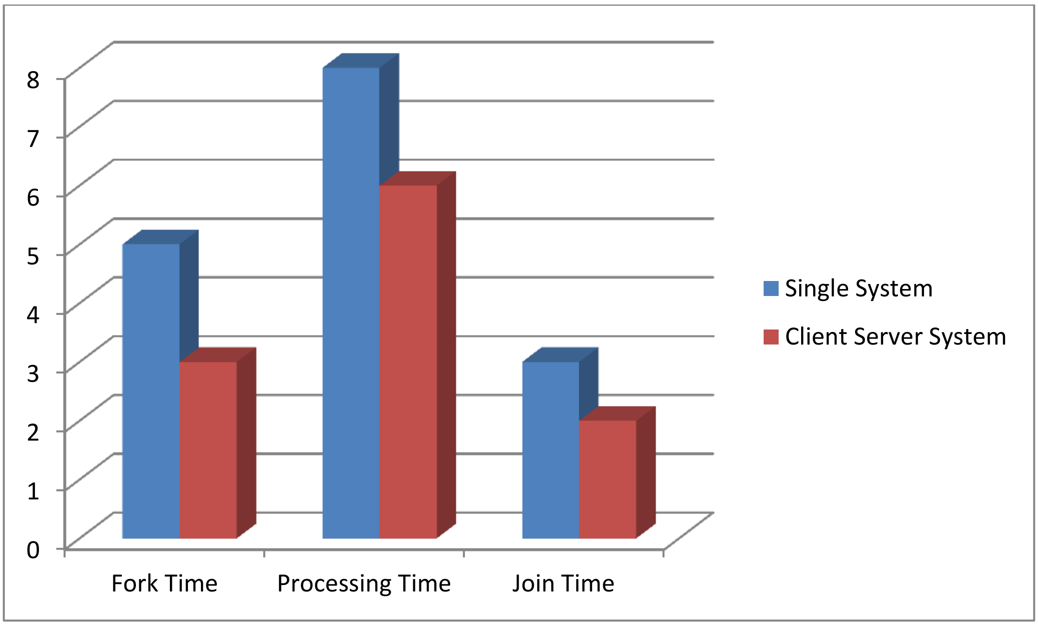

Figure 3: Barchart showing Comparison of single system and client server system

The system is also tested for various image sizes. Table 2 shows the analysis of various image sizes in single system.

Table 2 Processing in single system on various image sizes

Image Name	Image Size	Fork Time	Processing Time	Join Time	Total Processing Time	Efficiency	Speed	Overhead
Image1	512×512	5	8	3	16	86	71	63
Image2	256×256	4.5	7.7	2.8	15	88	73	60
Image3	128×128	4.1	7.3	2.5	13.9	89	76	58
Image4	64×64	3.8	6.9	2.0	12.7	90	79	55
Image5	32×32	3.5	6.5	1.8	11.8	91	81	53

From Table 2, it is studied that if the image is reduced, the processing time is also reduced. Even if the size of the image is reduced to half, the processing time is not reduced to half. The efficiency, speed and overhead is also reduced when image size is reduced.

Table 3 shows the Processing time in client-server system on various image sizes. The same images are chosen for single and client-server system.

Table 3 Processing in client-server system on various image sizes

Image Name	Image Size	Fork Time	Processing Time	Join Time	Total Processing Time	Efficiency	Speed	Overhead
Image1	512×512	3	6	2	11	92	85	48
Image2	256×256	2.8	5.6	1.9	10.3	93	86	46
Image3	128×128	2.6	5.4	1.7	9.7	94	88	44
Image4	64×64	2.4	5.2	1.5	9.3	94.5	89	42
Image5	32×32	2.2	5.0	1.3	8.7	96	92	40

In client server system also, the processing time is very much reduced if the image size is reduced.

V. CONCLUSION

In this work, a new system named Image Denoising based on Client-Server Architecture (IDCSA) is proposed. This system is based on ‘Divide and conquer’ methodology. Parallel implementation of algorithm is developed using matlab threads in order to leverage the parallel processing capability of current processors with multiple cores and we can see that the speed, efficiency, parallel time are good. We also focusd on other image parameters and results are evaluated. The performance of IDCSA is compared with several existing systems and the performance results of IDCSA are better. The execution time of this system is appreciable.

References:

[1] Rohit Verma, Dr. Jahid Ali, "A Comparative Study of Various Types of Image Noise and Efficient Noise Removal Techniques", International Journal of Advanced Research in Computer Science and Software Engineering, 3(10), pp. 617-622, 2013.
[2] Charles Boncelet, “Image Noise Models”. Alan C.Bovik. Handbook of Image and Video Processing, second Edition,2005.
[3] Reza Ahmadi., Javad Kangarani Farahani, Farbod Sotudeh, Ashkan Zhaleh, Saeid Garshasbi, “Survey of Image Denoising Techniques”, Life Science Journal, 10(1), pp.753-755,2013.
[4] Windyga, S. P, “Fast Impulsive Noise Removal”, IEEE transactions on image processing, 10(1), pp.173-178, 2001.
[5] Gholamreza Anbarjafari, Hasan Demirel, and Ahmet E. Gokus, "A Novel Multi-diagonal Matrix Filter for Binary Image Denoising", Journal of Advanced Electrical and Computer Engineering, 1, pp.14-21, 2014.
[6] Forest Agostinelli Michael R. Anderson Honglak Lee, "Adaptive Multi-Column Deep Neural Networks with Application to Robust Image Denoising", Advances in Neural Information Processing Systems, pp.1493- 1501, 2013.
[7] Preety D. Swami, Alok Jain, and Dhirendra K. Swami, "Region Classification Based Image Denoising Using Shearlet And Wavelet Transforms', International Journal of Computer Science and Information Technology, 6(1), pp.241-248, 2014.
[8] Punyaban Patel, Bibekananda Jena, Banshidhar Majhi, C.R.Tripathy, "Fuzzy Based Adaptive Mean Filtering Technique for Removal of Impulse Noise from Images", International Journal of Computer Vision and Signal Processing, 1(1), pp.15-21, 2012.
[9] Preeti kaur ,“Implementation of Image Processing Algorithms on The Parallel Platform Using Matlab”, International Journal of Computer Science & Engineering Technology (IJCSET), ISSN : 2229-3345 Vol.4 No.06, Page: 696-706, Jun 2013.