International Journal on Science and Technology (IJSAT)
E-ISSN: 2229-7677 ● Website: www.ijsat.org ● Email: editor@ijsat.org

"Time Optimization Strategies for Image Segmentation on Modern Multicore Architectures"

Dipika Sharma[1], Suraj Pal[2]

[1,2]Department of Computer Science, SBSSU, Gurdaspur
[1]sharma.dipika02@gmail.com, [2]Ersurajpal8@gmail.com

Abstract

Recent advancements in computer engineering have led to the development of multicore processors—chips that integrate two or more independent processing units (cores) within a single physical package. With growing demand for high-performance computing, multicore designs have become standard in modern processors. These architectures offer significant opportunities for performance gains, especially in fields like image processing. By adapting traditionally single-threaded code to run in parallel, developers can fully leverage the capabilities of multicore systems.

However, maximizing the performance of multicore architectures poses challenges. Beyond achieving raw speed, it is crucial to manage and utilize computing resources efficiently. Unlike older, single-core systems where performance improvements could be expected through hardware upgrades alone, modern applications must be redesigned using parallel algorithms to truly benefit from multicore systems. Image processing, which naturally involves tasks that can be executed concurrently, is particularly well-suited to parallelization. This dissertation focuses on harnessing the power of multicore systems for image processing tasks, with the primary goal of minimizing execution and wait times. To achieve this, we employed the fork-join model of parallel computing, enabling jobs to be executed simultaneously across multiple cores.

As part of this research, image segmentation algorithms were implemented using MATLAB. The study aims to improve overall system efficiency by reducing execution time and increasing responsiveness. Ultimately, the work demonstrates how parallel algorithm design can unlock the full potential of multicore processors in image processing applications.

1. Introduction

1.1 Distributed and parallel computing

Advancements in computer engineering have led to multicore processors—chips with multiple independent cores in a single package—becoming the standard in modern computing. This shift is driven by the demand for higher performance, particularly in areas like image processing. By adapting single-threaded programs to run in parallel, developers can harness the full potential of these systems.

However, utilizing multicore architectures effectively requires more than just parallel execution; efficient resource management is essential. Traditional sequential applications can no longer rely on hardware alone for performance gains. Instead, they must adopt parallel algorithms to capitalize on multicore capabilities. Image processing is especially well-suited to parallelization due to its inherent parallel structure. This dissertation explores how multicore systems can be leveraged to improve performance in image segmentation tasks. Using the fork-join parallel model and MATLAB implementations, the goal is to minimize execution and wait times, ultimately improving speedup and system responsiveness.[27]

1.3 Digital Image Processing

Digital image processing is the use of computer algorithms to perform image processing on digital images [26]. An image may be defined as a two-dimensional function, f(x, y), where x and y are spatial (Plane) coordinates, and the amplitude of at any pair of coordinates (x, y) is called the intensity or gray level of the image at that point. When x, y, and the amplitude values of f are all finite, discrete quantities, we call the image a digital image. The field of digital image processing refers to processing digital images by means of a digital computer. Note that a digital image is composed of a finite number of elements, each of which has a particular location and value. These elements are referred to as picture elements, image elements, peels, and pixels.

Pixel is the term most widely used to denote the elements of a digital image. Vision is the most advanced of our senses, so it is not surprising that images play the single most important role in human perception. However, unlike humans, who are limited to the visual band of the electromagnetic spectrum, imaging machines cover almost the entire spectrum, ranging from gamma to radio waves. They can operate on images generated by sources that humans are not accustomed to associating with images. These include ultra-sound, electron microscopy, and computer-generated images.

Thus, digital image processing encompasses a wide and varied field of applications. There is no general agreement among authors regarding where image processing stops and other related areas, such as image analysis and computer vision, start. Sometimes a distinction is made by defining image processing as discipline in which both the input and output of a process are images.

1.4Image Segmentation

Image segmentation is the process of partitioning a digital image into multiple segments. [1]. The goal of segmentation is to simplify or change the representation of an image into something that is more meaningful and easier to analyse. Image segmentation is typically used to locate objects and boundaries in images. More precisely, image segmentation is the process of assigning a label to every pixel in an image such that pixels with the same label share certain visual characteristics.

The result of image segmentation is a set of segments that collectively cover the entire image, or a set of contours extracted from the image. Each of the pixels in a region are similar with respect to some characteristic or computed property, such as colour, intensity, smoothness and no edge.

(a) (b)

Fig1.4: (a) original image (b) segmented image

2. LITERATURE SURVEY

In this section, work done in the area of image segmentation on single processor is reviewed and focus has been made on studying different algorithms for image segmentation. Performance of algorithm is main issue in image processing.

Agrawal, Pankaj, S. K. Shriwastava, and S. S. Limaye [1] (2016) presented the implementation using the GUI feature of the MATLAB and one best result is selected for any algorithm using the subjective evaluation. This implementation helped to find out the best suitable value of parameters for the segmentation of different types of imagery. One best algorithm has considered for each method of image segmentation. The interactive based method provides the facility to select the desired area as an object and produces better result. The proposed process also displayed the duration of segmentation of each algorithm.

Calderero, Felipe, and Ferran Marques [2] (2017) has proposed a family of unsupervised region merging techniques providing a set of the most relevant region-based explanations of an image at different levels of analysis. These techniques were characterized by general and nonparametric region models, with neither color nor texture homogeneity assumptions, nor a set of innovative merging criteria, based on information theory statistical measures. The scale consistency of the partitions is assured through i) a size regularization term into the merging criteria and a classical merging order, or ii) using a novel scale-based merging order to avoid the region size homogeneity imposed by the use of a size regularization term. It had made a complete and exhaustive evaluation of the proposed techniques using not only different databases for the two main addressed problems (object-oriented segmentation of generic images and texture image segmentation), but also specific evaluation features in each case: under- and over segmentation error, and a largest of region-based, pixel-based and error consistency indicators, respectively.

Chowdhury et. al [3](2019) reported method for image segmentation that combines region growing and edge detection. Existing schemes that use region-based processing provide unambiguous segmentation, but they often divide regions that are not clearly separated. Algorithm begins by using region growing to produce an over-segmented image. Then modify the over-segmented output of the region growing using edge criteria such as edge strength, edge smoothness, edge straightness and edge continuity. Two techniques - line-segment subtraction and line-segment addition - have been used. In the subtraction technique, the weakest edge is removed at each step. In addition, technique, the strongest edge is used to seed a multi-segment line that grows out from it at both ends. The addition technique produces better results than the subtraction technique where small curly regions produced by less significant regions in segmentation were not eliminated.

Felzenszwalb et. al [4] (2020) addressed the problem of segmenting an image into regions. A predicate was defined for measuring the evidence for a boundary between two regions using a graph-based representation of the image. An efficient segmentation algorithm was then developed based on that predicate, and showed that it produced segmentations that satisfy global properties. Algorithm was applied to image segmentation using two different kinds of local neighborhoods in constructing the graph, and illustrated the results with both real and synthetic images. Segmentation results of all the regions were combined to achieve the better segmentation. The run time of algorithm was nearly linear in the number of graph edges.

Gennart et. al [5] (2018) presented a tutorial description of the CAP Computer Aided Parallelization Tool. CAP has been designed with goal of letting the parallel application programmer having the complete control about how his application is parallelized. Authors discuss the issues of flow control and load balancing and show the solutions offered by CAP and also show how CAP can be used to generate relatively. Authors have also discussed how CAP can be used to generate relatively complex parallel programs incorporating neighborhood dependent operations.

Hsiao et. al [7] (2015) proposed a novel approach for edge-based image segmentation. Image segmentation and object extraction play an important role in supporting content-based image coding, indexing, and retrieval. Authors proposed an image segmentation algorithm by integrating mathematical morphological edge detector with region growing technique. The images are first enhanced by morphological closing operations, and then detect the edge of the image by morphological dilation residue edge detector. After region growing is implemented, homogeneous regions are merged together by applying the process of region merging algorithm.

3. PROBLEM FORMULATION

3.1 Problem definition

Parallel image processing offers significant advantages over its sequential counterpart, particularly in handling computationally intensive tasks. Utilizing multiple processing resources allows complex problems to be solved more efficiently and within shorter time frames. However, coordinating these resources and distributing the workload intelligently remains a key challenge for system designers. As user demands for increasingly complex and responsive applications grow, minimizing computation time while maintaining performance becomes critical.

This dissertation focuses on the effective use of multicore systems to address these challenges. The primary goal is to reduce both execution time and user wait time by implementing parallel processing techniques. Specifically, we employ the fork-join model of parallel algorithms, which enables tasks to be executed simultaneously across multiple cores, thereby maximizing processor utilization and speedup. To achieve the research objectives, image segmentation algorithms were implemented using MATLAB. Both sequential and parallel versions of the algorithms were developed to analyze performance differences. For parallel execution, we used MATLABPOOL, allowing parallel tasks to run concurrently across available cores.

To evaluate performance, we tested the algorithms on images of various sizes. Results indicate that as image size increases, so do speedup and efficiency—demonstrating the scalability of the parallel approach. Additionally, several image quality metrics were analyzed to assess the effectiveness of the proposed segmentation algorithm.

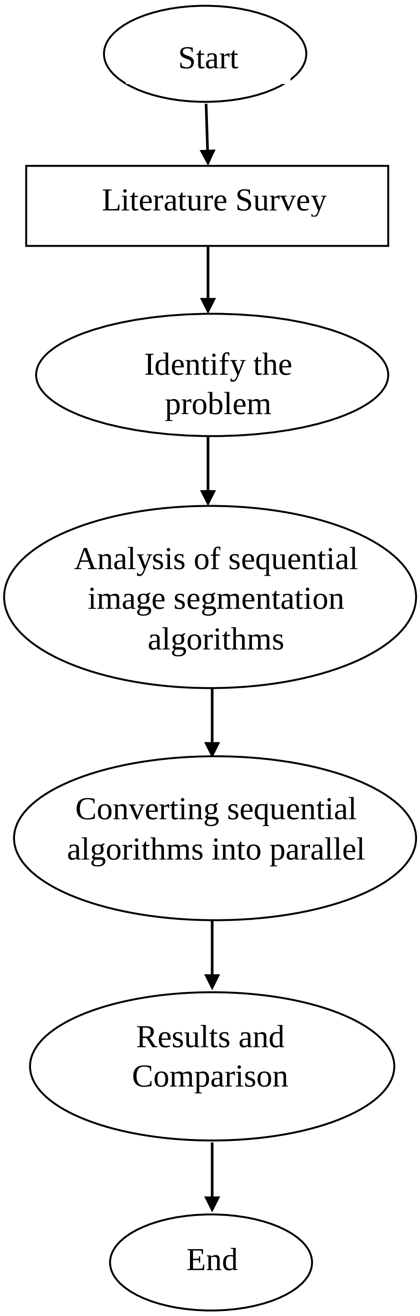
3.2 Objectives

The objectives of this dissertation are:

1. The main objective is to reduce the amount of required time to segment the digital images.
2. To obtain good load balancing among all the resources of the system in terms of number of tasks scheduled on each resource.
3. This research work deals with the process to collect, analyse and evaluate the digital images to prove the effectiveness and efficiency of proposed strategy.
4. The objective of work is to maximize multicore utilization for image segmentation based parallel algorithms.

3.3 Methodology

In order to improve the speedup and efficiency by reducing the time of execution after implementing in parallel manner, the following methodology is used.



4. RESULTS AND DISCUSSIONS

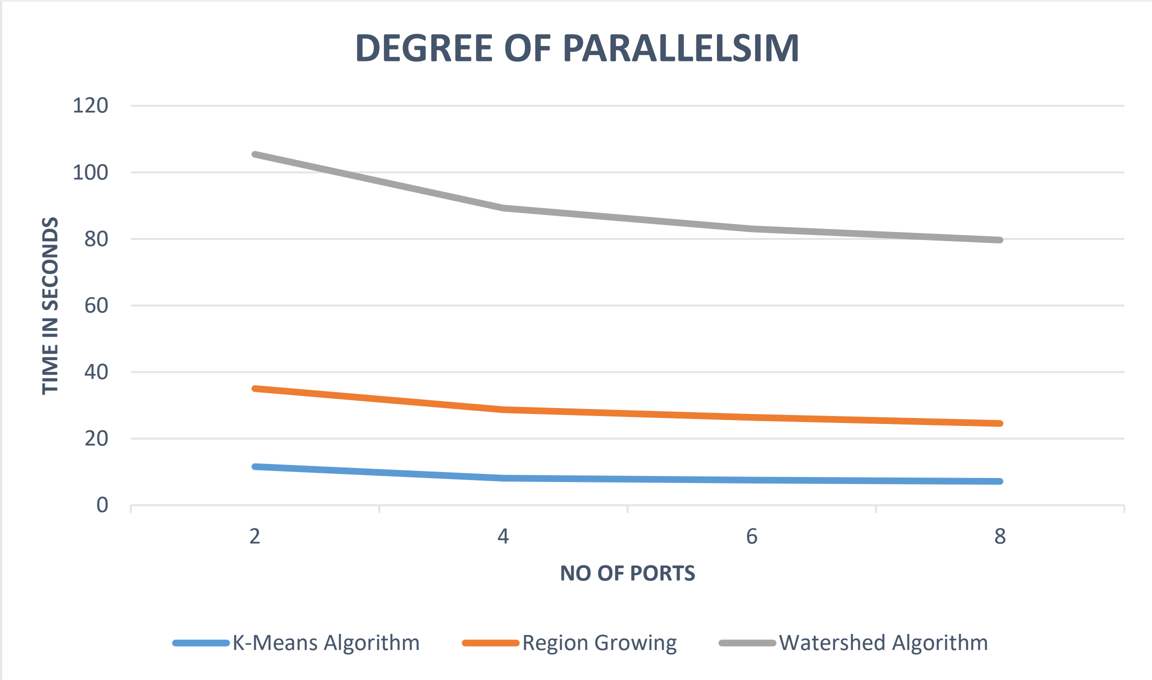


Fig 4.5.1: Degree of Parallelism

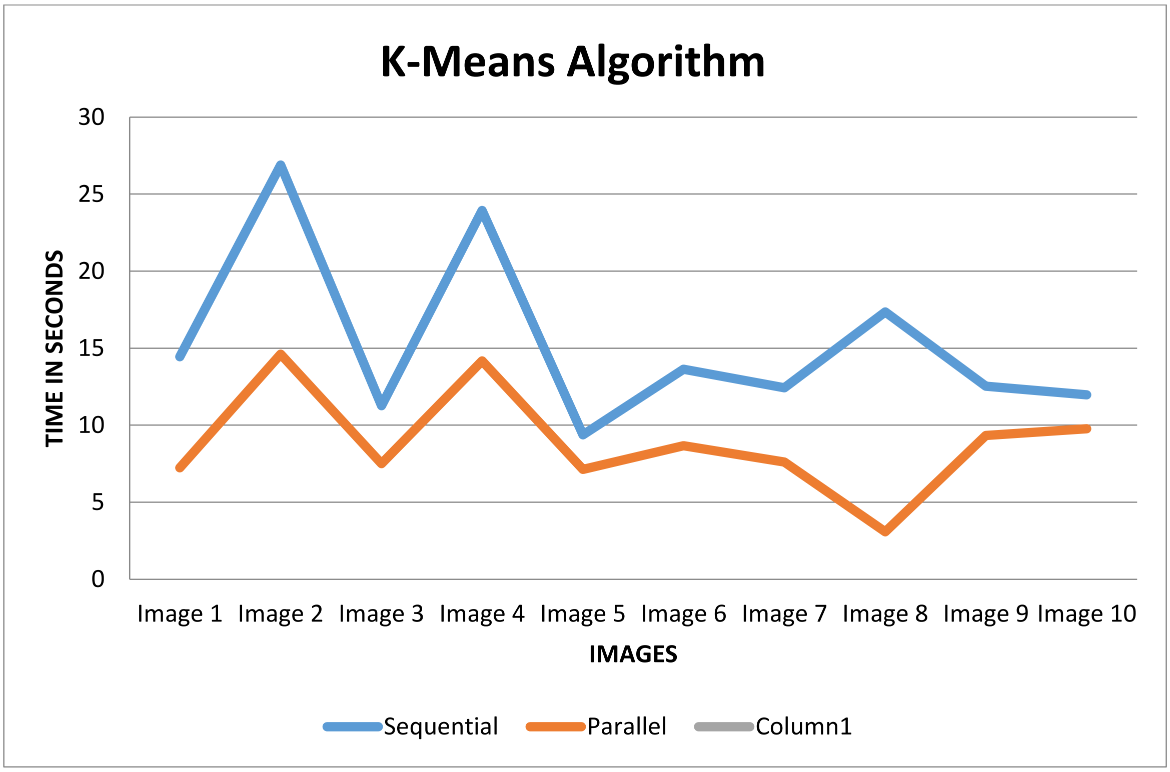


Fig 4.5.2 Execution time of K-Means Algorithm on all images.

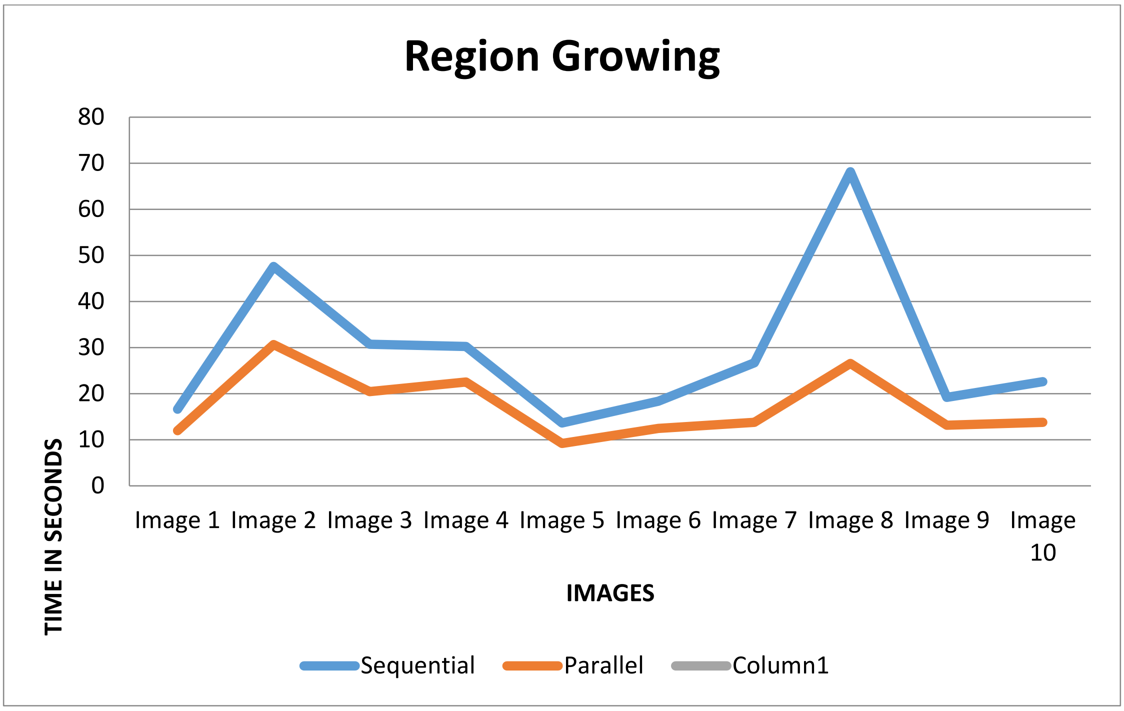


Fig 4.5.3 Execution time of Region Growing on all images

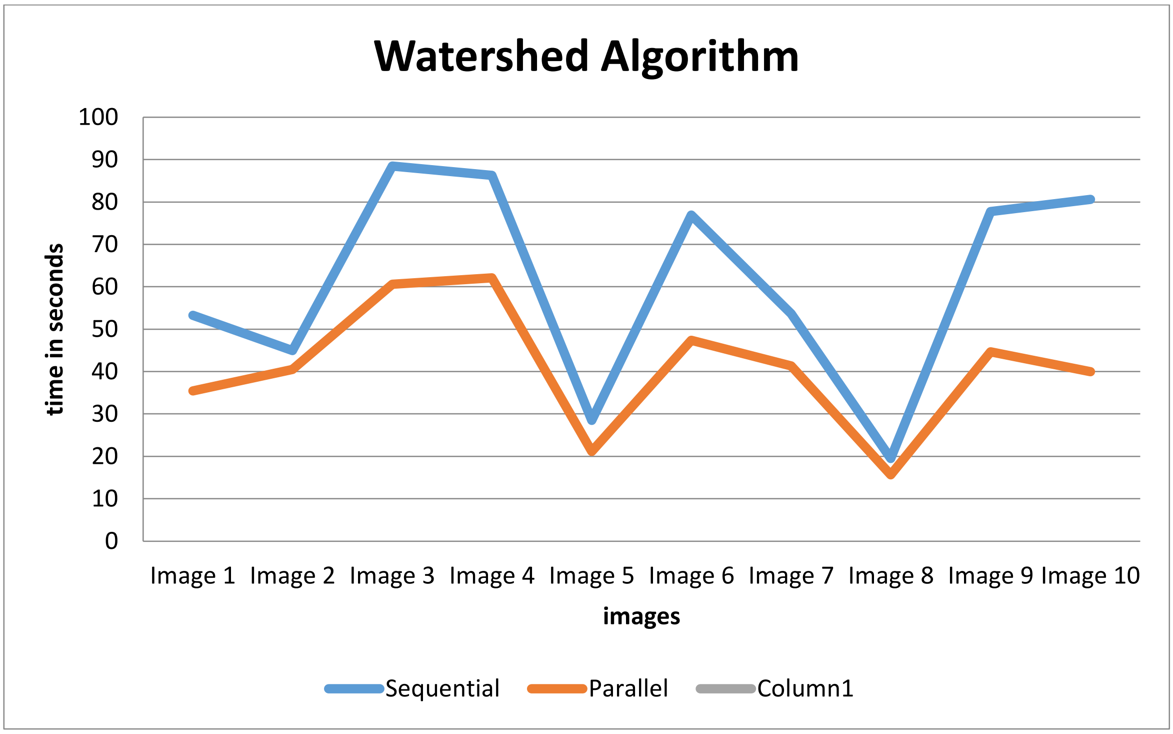


Fig 4.5.4 Execution time of Watershed Algorithm on all images.

4.5.3 PSNR

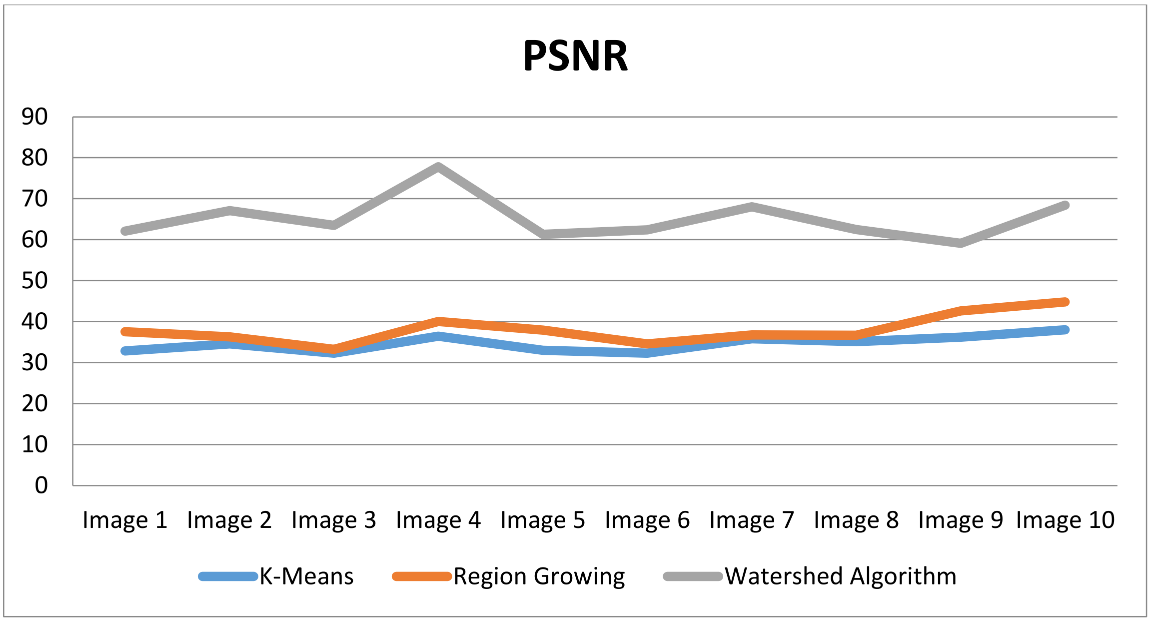


Fig 4.5.5 shows peal signal to noise ratio (PSNR) of all images

4.5.4 MSE

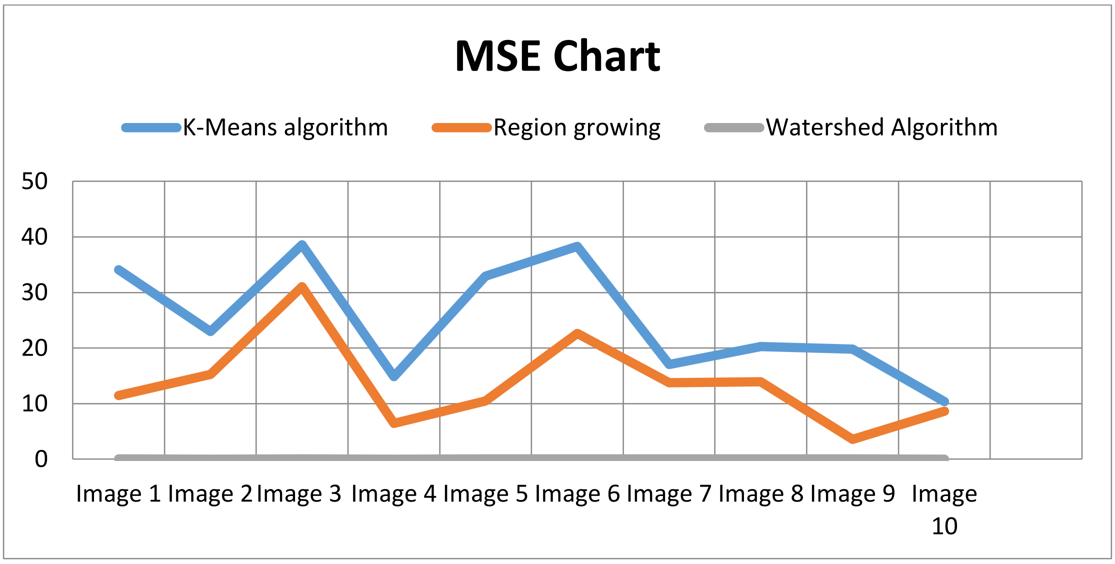


Fig 4.5.6shows mean square error (MSE) of all images

4.5.5 RMSE

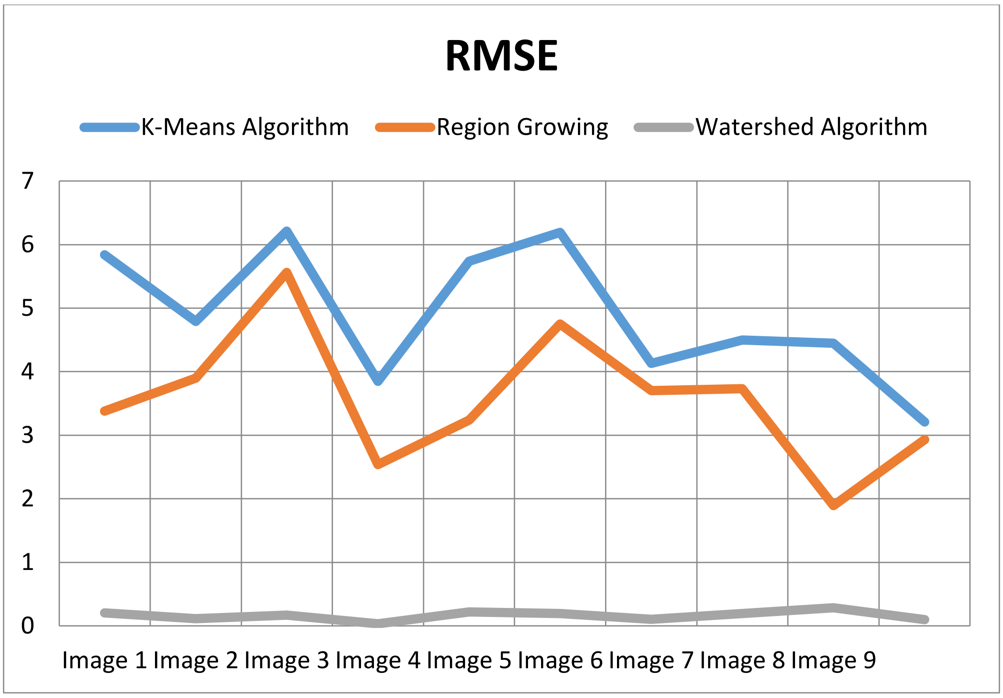


Fig 4.5.7 shows root mean square error (RMSE) of all images

5. CONCLUSION AND FUTURE SCOPE

Image segmentation is one of the important operations on a digital image to represent the objects in the image. This work presents parallel implementation of sequential image segmentation algorithm. Parallel implementation of algorithm is done using MATLAB threads in order to leverage the parallel processing capability of current processors with multiple cores and results prove that the speed up, efficiency, parallel time that is computed is good. Other image parameters are also focused and results are evaluated. The focus of this implementation is to improve the performance of image processing algorithm and maximum utilization of the multicores. In terms of performance, parallel implementation is about 1.5 times faster than the sequential processing. This is a very promising result since it allows the exploitation of the vast processing power of current processors with multiple cores.

In the future, the intention is to use the same principle of division of work in tiles to calculate more and more parameters and proposed the ways to utilize resources better.

REFERENCES

1. Agrawal, Pankaj, S. K. Shriwastava, and S. S. Limaye. "MATLAB implementation of image segmentation algorithms." In Computer Science and Information Technology (ICCSIT), 2016 3rd IEEE International Conference on, vol. 3, pp. 427-431. IEEE, 2016.
2. Calderero, Felipe, and Ferran Marques. "Region merging techniques using information theory statistical measures." Image Processing, IEEE Transactions on 19, no. 6 (2017): 1567-1586.

3. Chowdhury, Mahbubul Islam, and John A. Robinson. "Improving image segmentation using edge information." In Electrical and Computer Engineering, 2019 Canadian Conference on, vol. 1, pp. 312-316. IEEE, 2019.
4. Felzenszwalb, Pedro F., and Daniel P. Huttenlocher. "Efficient graph-based image segmentation." International Journal of Computer Vision 59, no. 2 (2020): 167-181.
5. Gennart, Benoit A., and Roger D. Hersch. "Computer-aided synthesis of parallel image processing applications." In SPIE's International Symposium on Optical Science, Engineering, and Instrumentation, pp. 48-61. International Society for Optics and Photonics, 2018.
6. Gonzalez, Rafael C., Richard E. Woods, and Steven L. Eddins. "Digital image using Matlab processing." Person Prentice Hall (2019): 183-193.
7. Hsiao, Ying-Tung, Cheng-Long Chuang, Joe-Air Jiang, and Cheng-Chih Chien. "A contour based image segmentation algorithm using morphological edge detection." In Systems, Man and Cybernetics, 2015 IEEE International Conference on, vol. 3, pp. 2962-2967. IEEE, 2015.
8. Huang, Chaobing, Quan Liu, and Xiaopeng Li. "Color image segmentation by seeded region growing and region merging." In Fuzzy Systems and Knowledge Discovery (FSKD), 2017 Seventh International Conference on, vol. 2, pp. 533-536. IEEE, 2017.
9. Huang, Yong-Ren, and Chung-Ming Kuo. "Image segmentation using edge detection and region distribution." In Image and Signal Processing (CISP), 2019 3rd International Congress on, vol. 3, pp. 1410-1414. IEEE, 2019.