MIDDLE EAST UNIVERSITY
جامعة الشرق الأوسط
Amman - Jordan

Efficient Graph-Based Image Segmentation for Natural Images

تقسيم الصور الطبيعية الفعال باستخدام التمثيل البياني

By

Laila Khalil Almugheer

Supervisor
Dr. Ahmad Adel Abu-Shareha

Thesis Submitted In Partial Fulfillment of the Requirements for the Degree of Master of Computer Science

Department of Computer Science

Faculty of Information Technology

Middle East University

June- 2017

Authorization

I, Laila Khalil Almugheer, authorize Middle East University (MEU) to provide copies of my thesis to the concerned libraries, establishments, and institutions upon request.

Name: Laila Khalil Almugheer

Date: 04-06-2017

Signature:

اقرار تفويض

انا ليلى خليل المغيرافوض جامعة الشرق الاوسط بتزويد نسخ من رسالتي للمكتبات المعنية، المؤسسات، الهيئات عند طلبها.

الاسم: ليلى خليل المغير

التاريخ: 4-6-2017

التوقيع:

Examination Committee Decision

This is to certify that the thesis entitled "Efficient Graph-Based Image Segmentation for Natural Images" was successfully defended and approved on 4-6-2017.

Examination Committee Members	Signature
(Supervisor) **Dr. Ahmad Abu Shariha** *Assistant Professor* *Middle East University*	
(Chairman of Examination Committee) **Dr. Fayez Alshrouf** *Associate Professor* *Middle East University*	
(External Committee Member) **Prof. Jehad Al-Saadi** *Professor* *Arab Open University*	

ACKNOWLEDGEMENT

In the name of Allah, the Most Gracious and the Most Merciful Alhamdulillah, all praises to Allah for the strengths and His blessing in completing this thesis.

Special appreciation goes to my supervisor, Dr. Ahmad Adel Abu-Shareha, for his supervision and constant support. His invaluable help of constructive comments and suggestions throughout the thesis works have contributed to the success of this research. Not forgotten, my appreciation to all my colleagues for the friendship and memories. Last but not least, my deepest gratitude goes to my beloved parents; Mr. Khalil Almugheer and Mrs. Mouroge Hindawi and also to my sisters and brothers for their endless love, prayers, encouragement love and care. To those who indirectly contributed in this research, your kindness means a lot to me. Thank you very much.

DEDICATION

فَتَعَالَى اللَّهُ الْمَلِكُ الْحَقُّ ۗ وَلَا تَعْجَلْ بِالْقُرْآنِ مِن قَبْلِ أَن يُقْضَىٰ إِلَيْكَ وَحْيُهُ ۖ وَقُل رَّبِّ زِدْنِي عِلْمًا

سورة طه – الآيه 114

I dedicate this work first, to my beloved family; you are the real blessings in my life

To My Father, for his support and encouragement.

To My Mother, for her deep love, you are the Source of my strength and inspiration.

To My Brothers, you both are my backbone.

To My Sisters, your love means a lot to me.

To My Relatives and Friends, for their help and sweet words.

Last but not least, to every woman out there; you are capable to success and achieve your dreams, just believe in yourself.

Table of Contents

Content **Page Number**

List of Tables

List of Figures

List of Abbreviations

RGB:	**Red – Green – Blue**
HSI:	**Hue – Saturation – Intensity**
PB:	**Probability of Boundary**

Efficient Graph-Based Image Segmentation for Natural Images

By

Laila Khalil Almugheer

Supervisor

Dr. Ahmad Adel Abu-Shareha

Abstract

Image segmentation is the process of partitioning an input image into multiple segments (sets of pixels, also known as super-pixels). The widespread utilization of segmentation in various real-time applications has urged the need to reduce the segmentation cost without compromising the efficiency of the produced output. Graph-based is one of the effective techniques for image segmentation. Although, graph-based uses a greedy approach to illuminate connected edges between distinctive regions in several iterations, it produces satisfactory results compares to the existing segmentation approaches. However, graph-based technique consumes time, which make it inapplicable for real-time applications.

This thesis proposed an extended graph-based image segmentation technique by reduce the segmentation time of the original technique and improve the accuracy of the output results. To achieve the aforementioned goal, the process, which is implemented over a constructed graph corresponding to an input image, forms homogenous regions around initial points that are selected automatically. The initial points, which characterized as significant spots in the image that lies within a coherent region, are determined based on the intensity of the neighborhood pixels. By using initial points, the number of iterations that is required to complete the segmentation process is decreased, which will decrease the running time. Moreover, based on the initial points, the regions that are produced in the segmentation process are coherent since it is accumulated over coherent pixels, which will enhance the accuracy of the output results. The experiments were conducted on Pascal-challenge images. The results showed that the proposed segmentation technique gain an accuracy of 84%, which is higher than the original approach that achieved an accuracy of 68%. Moreover, the proposed segmentation approach reduces the running time of the segmentation by 62% compared to the original approach.

Key Words: Image Segmentation, Image Processing, Initial Points Selection.

تقسيم الصور الطبيعية الفعال باستخدام التمثيل البياني

إعداد

ليلى خليل المغير

إشراف

الدكتور أحمد عادل أبو شريحة

الملخص

يعد تقسيم أو تجزئة الصور الى مناطق متعددة من التقنيات المهمة في معالجة الصور. وقد الحّ الاستخدام المتزايد للتجزئة في مختلف تطبيقات الوقت الحقيقي في الحاجة إلى خفض تكلفة التجزئة دون المساس بكفاءة الناتج. تجزئة الصور على أساس الرسم البياني هو أحد التقنيات الفعالة لتقسيم الصور، تقوم هذه التقنية بتحويل الصورة إلى رسم بياني غير موجه. على الرغم من أن تقسيم الصور على اساس الرسم البياني يستخدام نهج جشع لازالة الحواف المتصلة بين المناطق المختلفة من خلال فحص الحواف عدة مرات، إلا أن نتائجها مرضية مقارنة مع مغيرها من المناهج المستخدمة لتقسييم الصور. ورغم النتائج الجيدة إلا أن هذه التقنية تستهلك وقتا طويلاً، مما يجعلها غير قابلة للتطبيق في للتطبيقات الوقت الحقيقي.

يهدف العمل المقترح في هذه الأطروحة إلى تحسين الية تجزئة الصور القائمة على الرسم البياني من خلال تقليل وقت تجزئة وتحسين دقة نتائج . لتحقيق الهدف المذكور أعلاه، يتم تنفيذ تجزئة الصورة بواسطة تحويل الصور إلى رسم البياني ومن ثم تطبيق التقسيم بالرجوع الى نقاط أولية، وهي نقاط هامة في الصورة تقع داخل منطقة متجانسة، يتم تحديد النقاط الأولية أولا متبوعة بتنفيذ عملية التقسيم مستندة إلى الرسم البياني بتراكم وحدات البكسل حول هذه النقاط. باستخدام النقاط الأولية سيتم تنفيذ التكرارات على عدد محدود من وحدات البكسل في الصورة، مما سيقلل من

وقت التنفيذ. وعلاوة على ذلك، واستنادا إلى النقاط الأولية، تكون المناطق التي تنتج في عملية التجزئة متجانسة لأنها تراكمت على وحدات متجانسة، مما سيعزز دقة نتائج المخرجات.

أجريت التجارب على صور باسكال، وأظهرت النتائج أن العمل المقترح قد حقق دقة 84٪، وهي نسبة أعلى من المنهج الأصلي الذي حقق دقة 68٪ وعلاوةً على ذلك، فإن العمل المقترح قد قلل من وقت التنفيذ بنسبة 62٪ مقارنة بالنهج الأصلي.

الكلمات المفتاحية: تجزئة الصورة، معالجة الصورة، اختيار النقاط الأولية .

Chapter One

Introduction

Image is an optical counterpart or appearance of an object that is produced by mirror reflection, lens refraction or the passage of luminous rays through a small aperture and their reception on a surface. Nowadays, images are used in each part of our life, such as: person identity, captured memories, medical diagnosis, evidence, and so on. Surely, it will not be efficient to use physical images to store this huge number of images. The new technology of digital images and its associate processing techniques, such as graph-based segmentation solves many issues with photographic images, such as long-term storage, ease access, enhancement and information extraction. In this chapter, the significant of the proposed work over graph-based image segmentation is highlighted and justified.

1.1 Background

Digital image is numeric two-dimensional representation of an image in the machine in a form of an array of vector or raster. Vector is used with fixed resolution images and raster is used with infixed resolution (Sarfraz, 2004). By default, the term "digital image" refers to a raster type of image. Digital image is created through the process of digitization, which is the process of transforming images, text, or sound from analog media into electronic data (Bellis, 2016). Digital media ease the tasks of saving, organizing, retrieving, processing and restoring contents using electronic devices (Feinberg et al., 2016).

Image processing is the set of techniques that analyze, manipulate and improve the quality of the digital images. The application of image processing is growing rapidly because it improves information gathering, automatic recognition of objects in a scene, and so on (Umbaugh, 2010). Digital image processing mainly uses computer algorithm and mathematical equations on the pixels of the image. These algorithms allow sophisticated operations and serves many real-life applications, that includes but not limited to: medical field, remote sensing, transmission and encoding, machine/robot vision, color processing, pattern recognition, video processing, and microscopic imaging (Szeliski, 2010). Image processing is categorized into various groups, each implements different tasks and obtained unique results. Overall these groups share the same goals, image enhancement and information extraction (Dagar & Dahiya, 2016). Figure 1.1 illustrates an example of an output obtained by an image processing technique, which is image sharpening and restoration., while Figure 1.2 illustrates an example of object recognition.

Before After

Figure 1.1: Example of Image Sharping and Restoration

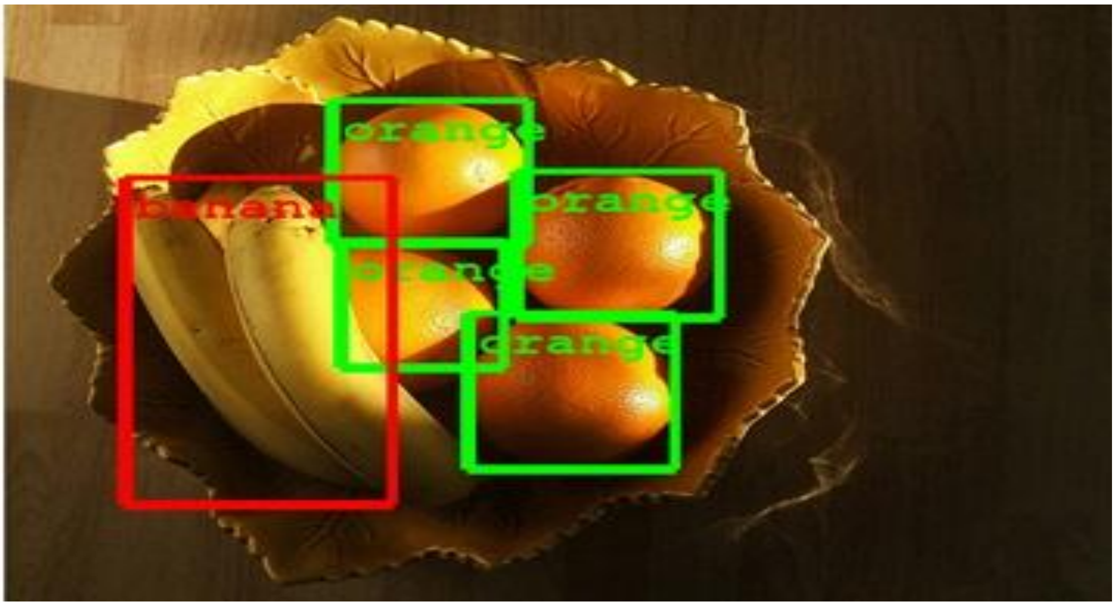
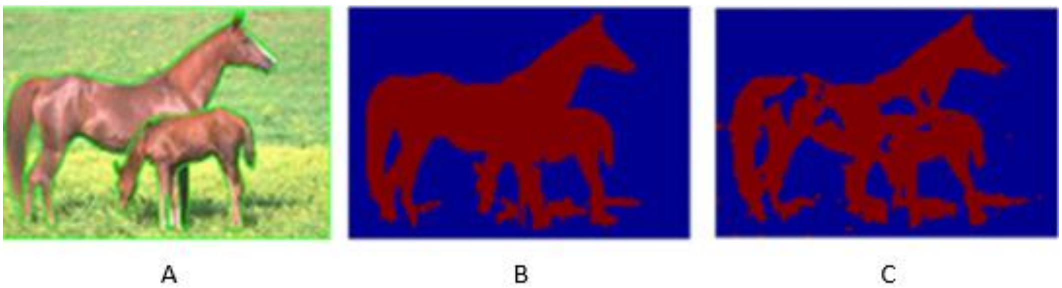


Figure 1.2: Example of Object Recognition

Image segmentation, is an important technique in image processing. Segmentation is the process of partitioning a digital image into multiple segments (sets of pixels, also known as super-pixels). In a segmented image, each segment usually refers to a meaningful shape or item. The goal of segmentation is to simplify and/or change the representation of an image into something that is more meaningful and easier to analyze. Segmentation can be done according to various segment features, such as brightness, color, or motion using various techniques (Saini, 2014). Figure 1.3 shows an example of the output of an image segmentation process. As illustrated the segmented images clearly identified the objects in the underlying image.



(A) The original image, (B) Segmented image, (C) Segmented image

Figure 1.3: Example Segmentation Output

Image segmentation is an important approach of image processing and image analysis. The more reliable segmentation leads to more successful image processing, advanced information extraction and better image understanding. After segmentation is done, information about objects defines by regions can be easily extracted. Information gathered may include object type, orientation, size, etc., Such information is valuable and essential in various fields, such as medical field, remote sensing field and so on (Glasbey,1995). Image segmentation is widely used for image recognition and image compression, since it is neither efficient nor practical to start compression on whole image directly, while performing image segmentation is to classify or cluster an image into several parts (regions) per the features of the image, therefore result in more efficient compression. Other applications that uses segmentation are: object recognition, image compression, content based image retrieval, market analysis, machine learning and computer vision (Zhang et al., 2016).

The increasing utilization of segmentation in various real-time applications has urged the needs for reducing the segmentation cost without compromising the efficiency of the produced output. Graph-based is one of the effective techniques for image segmentation. This technique transfers the image into undirected graph, in which the nodes are corresponding to pixels in the image, edges corresponding to the connectivity between pixels and the weight of the edges are calculated based on the difference in intensity between each neighborhood pixels. Although, graph-based uses a greedy approach to illuminate connected edges between distinctive regions in several iterations, it produces satisfactory results compares to the existing segmentation approaches. However, graph-based technique consumes time, which make it inapplicable for real-time applications.

1.2 Problem Statement

As mentioned, there is a need to improve the segmentation accuracy and reduce its time consumption to facilitate the utilization of segmentation in various real-life applications, such as medical imaging, remote sensing and etc. Graph-based technique, as one of the most utilized segmentation techniques, was proved to provide satisfying results, yet it consumes time. Subsequently, it is inefficient to use graph-based segmentation with huge dataset of images with high resolution. Subsequently, in thesis, the problem of reducing the time consumption in the graph-based image segmentation while preserve the segmentation quality is investigated. However, in order to maintain the accuracy of the output results, the process not be modified. Instead, the number of processing iteration, in graph-based segmentation, shall be reduced without affecting the overall mechanism that is built over a graph representation. This problem can be further divided into the following sub-problems:

- How to modify graph-based segmentation technique and decrease the number of iterations while maintain the accuracy of the output results by maintaining the core process of the original technique.

- How to form the segmentation process around selected homogenous pixels, instead of iterating over all the pixels in the image.

- How to automatically select homogenous pixels in the image, which is caled initial points, by which the iterations can be decreased.

- How to implement the process with predefined initial points and evaluate the results using an image dataset with known ground-truth segmentation.

1.3 Goal and Objectives

Graph-based image segmentation depends examining the weights of the arcs for possible region splitting or region merging at each examined arc. This process is implemented in several iterations that keeps on refining regions by splitting and merging boundaries by examining the all the arcs over and over. Obviously, there is a potential opportunity in reducing the process by avoid the examination the relationships of connected pixels that are obviously falls into same/different regions. By implementing such reduction, a major benefit is acquired by reducing the number of iteration required as the number of involved pixels is reduced. The goal of this thesis is to resolve this issue by defining initial points in the image and applies the algorithm depending on these points and to use the existing techniques for initial point selection as has been reported in the literature with region-growing segmentation techniques. The objectives in this research are:

- To modify the process of graph-based segmentation technique in order to decrease the number of examined arcs and the number of iterations performed while maintain the core segmentation process.

- To form the segmentation process around initial points with specific homogenous characteristics.

- To automatically define significant initial points in the image and use these points to guide the merging process in the graph-based algorithm.

- To implement, evaluate and compare the results of the proposed work using an image dataset with known ground-truth segmentation.

1.4 Motivations

The usage of the internet technologies and search engines, increase the need to automatically analyze images, which makes image processing a very important and useful field. Subsequently, reducing the time consumption of complicated image processing techniques has motivated researchers in image processing field in the last decades. Similarly, the proposed work is motivated with the need to reduce the time consumption of the graph-based image segmentation. This motivation can be summarized in the following points:

- There is a potential opportunity in obtained an accurate segmentation output using the graph-based image segmentation technique.

- There is a need to improve graph-based image segmentation accuracy and reduce its time consumption to facilitate the utilization of segmentation in various real-life applications, such as medical imaging, remote sensing and etc.

- There is a potential opportunity in reducing the process of graph-based image segmentation by avoid the examination the relationships of connected pixels that are obviously falls into same/different regions.

- There is a potential opportunity to reduce the number of iteration required to achieve the segmentation task as the number of involved pixels is reduced.

- There is a potential opportunity in using automatic initial point selection techniques that are used mainly with region-growing image segmentation.

1.5 Limitation

Since digital images can be categorized into several categories like Natural, graphical, virtual, etc., it is sufficient to work on a specific image category. This work will address image processing of natural images using graph-based image segmentation.

1.6 Thesis Organization

This thesis is organized into five chapters. **Chapter One** introduced the principles behind graph-based image segmentation along with the research problem, objectives and contribution.

Chapter Two presents the literature review of the image segmentation techniques and their advantages, disadvantages and characteristics.

Chapter Three covers the proposed work that converts image into two-dimensional graph and calculates the edges weight based on the difference between the connected pixels. Then, the proposed work defines initial points to run segmentation technique around them.

Chapter Four covers the implementation, analysis and discussion of the output results as well as its performance through detailed experiments.

Finally, **Chapter Five** provides the conclusions and recommendations for further research.

Chapter Two
Literature Review and Related Work

In this chapter, literature reviews on image segmentation techniques, image segmentation classification, advantages and disadvantages are given. Moreover, graph-based image segmentation is discussed in details and the related works on enhancing and improving this technique are discussed accordingly. This chapter gives a review on segmentation and graph-based segmentation in order to situate the graph-based technique among the existing segmentation literature. The organization of this chapter is as follows: Section 2.1 is an introduction to the segmentation process and techniques. Section 2.2 reviews the different types of image segmentation. Section 2.3 discusses the related work. Finally, a summary is given in Section 2.4.

2.1 Introduction

Various techniques for image segmentation was proposed in the literature, some of these techniques were developed to segment specific types of images, such as segmenting medical images, and some others were proposed to segment all types of images. Among the existing techniques, graph-based image segmentation is a general robust technique that was built with the notion of graphs and graph cuts.

The segmentation techniques, was originally built based on theories from other iterative and non-iterative data-processing techniques, such as clustering and classification. Subsequently, there is a great similarity between data clustering and image segmentation. Notably, part of the literature will cover the clustering techniques that helped and can be used for image segmentation.

2.2 Segmentation Techniques

Because of its variety, various classifications of the exiting segmentation techniques are possible, as illustrated in Figure 2.1. One of the visible classification is the one based on the context, while the other is based on the utilized technique, which classify the underlying approaches into: Clustering, Edge linking, and Region Operation (Semwal et al., 2016).

2.2.1 Context Segmentation

Based on the context, segmentation techniques can be classified into contextual segmentation and non-contextual segmentation.

2.2.1.1 Non-Contextual Segmentation

A non-contextual segmentation technique uses single separation point called a threshold to segment the image into at least two regions, one with pixels that have values below the threshold, other with pixels that have values equals to or higher than the threshold.

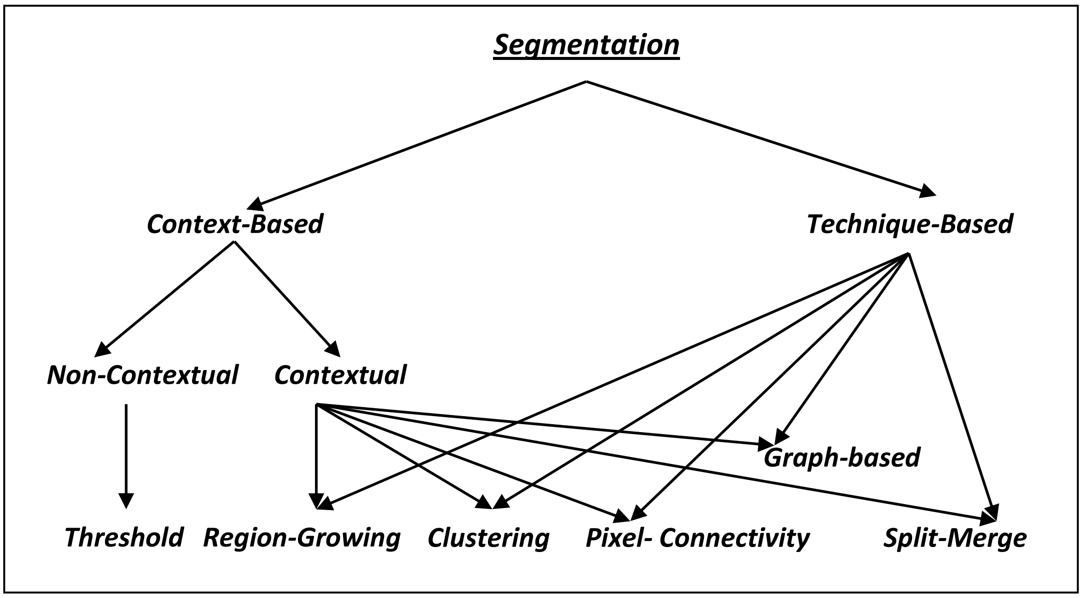


Figure 2.1: Classification of the Segmentation Approaches

The non-contextual segmentation does not take into consideration the contextual information of the pixels (location and neighborhood) and the output results are varied according to the selected threshold type and value, as illustrated in Figure 2.2 (Mottaghi et al., 2014). Various non-contextual segmentation techniques were proposed, such as the simple threshold, the adaptive threshold and color threshold. Each of these techniques uses different image information or generates different numbers of output segment. Simple threshold segments gray level images into a background and image segments. The histogram of an image is constructed, in simple threshold technique, and the peaks of the histogram are identified. Then, these picks are used as the components of different image regions. The two highest peaks are used as the component of the segments/regions and the middle point between these peaks are used as the threshold. Figure 2.3 illustrates an example of segmentation using simple threshold technique with different threshold value and subsequently different results (Vala & Baxi, 2013).

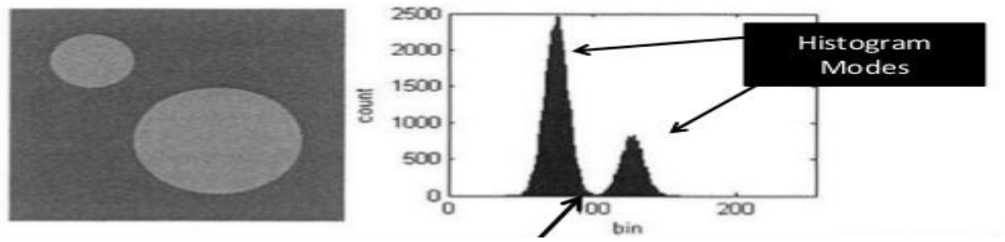


Figure 2.2: Non-Contextual Segmentation

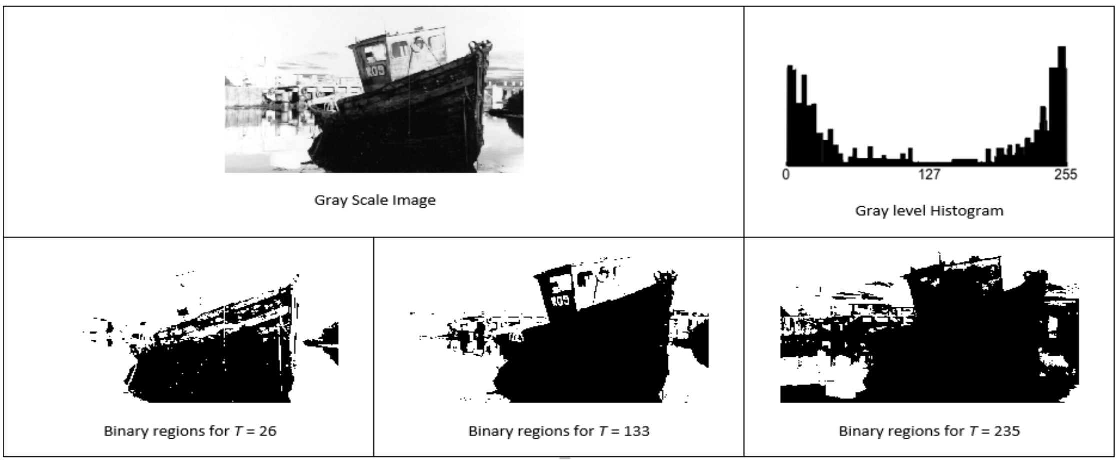


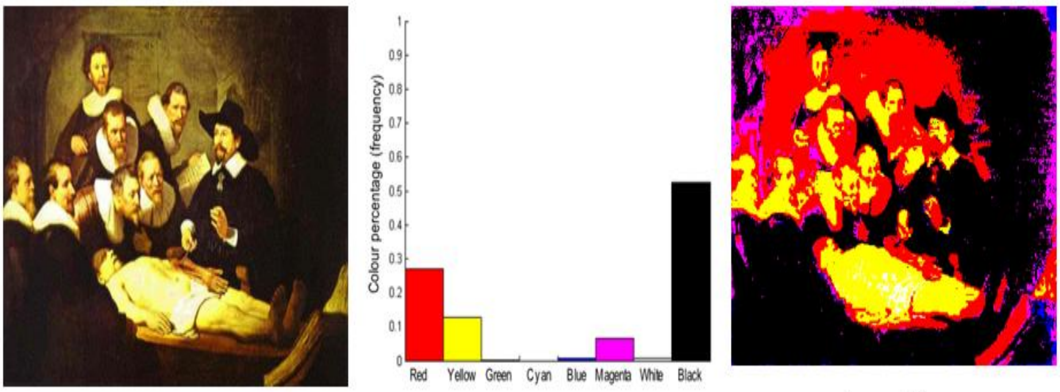
Figure 2.3: Simple Threshold Example

Adaptive threshold is similar to the simple threshold but using extra features. Adaptive threshold uses more than one threshold value. The threshold value changes dynamically to accommodate the change in lighting levels in the image (Issac & Dutta, 2015).

Color threshold uses color system, such as RGB or HIS instead of the gray scale. The color threshold is meant to produce more accurate segmentation since colored pixels provide more reliable information. Figure 2.4 illustrates an example of segmentation using color threshold technique (Rungruangbaiyok, & Chetpattananondh, 2015).

2.2.1.2 Contextual Segmentation

Contextual segmentation technique, unlike the non-contextual segmentation, takes into consideration the location of the pixels. This technique finds similarity within the same region and discontinuity between regions, which defines the objects boundaries. Various contextual segmentation techniques were proposed, such as the pixel connectivity, region growing and split and merge techniques (Subasic, 2014).



Original Colored Image Histogram Segmentations

Figure 2.4: Color Threshold Example

Pixel connectivity is based on pixel neighborhoods system that may consists of four connected pixels (up, down, right, and left) or more advanced system that uses eight connected pixels (up, down, right, left, and diagonal pixels), as illustrated in Figure 2.5.

Pixel and its connected pixels are tested against a threshold, to define which pixels are on the same region. Then, each connected pixel with its neighborhood pixels are tested without redundancy. This process continue until no more related connected pixel is defined for the same region. Figure 2.6 illustrates an example of segmentation using pixel connectivity technique (Rudra et al., 2013).

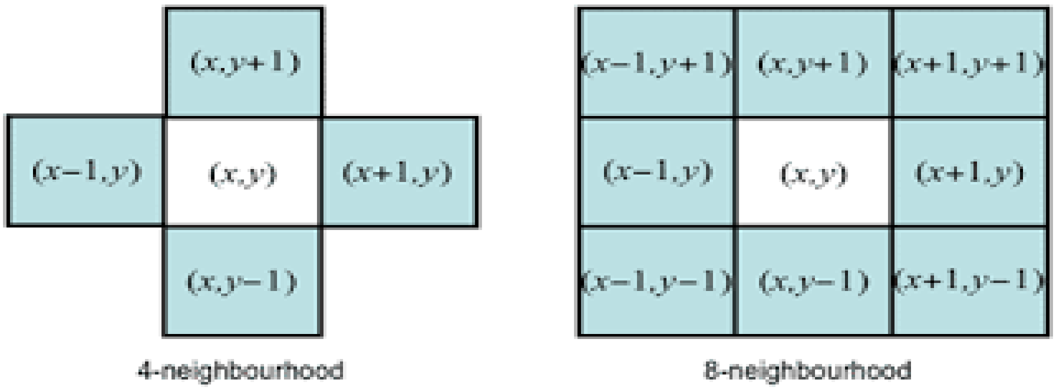


Figure 2.5: 4-Connected vs. 8-Connected Neighboring Systems

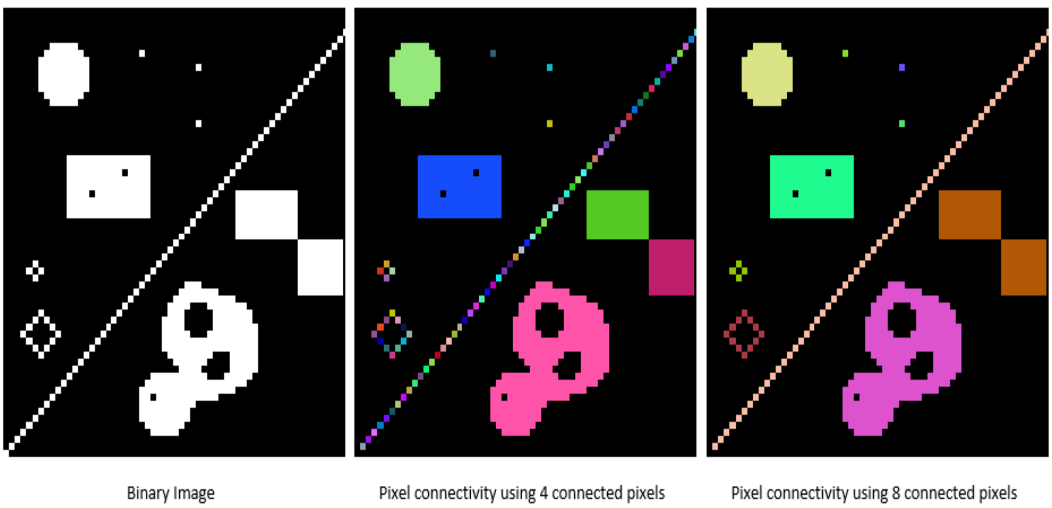


Figure 2.6: Pixel Connectivity Segmentation Example

Region growing uses user's select seeds, or in more advanced techniques, the seeds are chosen automatic. Then, a bottom up algorithm is applied on the selected seeds to find neighborhood pixels with similar features to form a region, if and only if it was not added to another region. This technique obtained good results but it is not stable, since the results would change if the pre-selected seeds are changed, or if the seeds selection is not accurate. Figure 2.7 gives illustration of the growing process. Figure 2.8 shows an image of peacock where single seed is chosen so the result contain two regions: one for region around the seed and other for the rest of image representing back ground (Holz & Behnke, 2014).

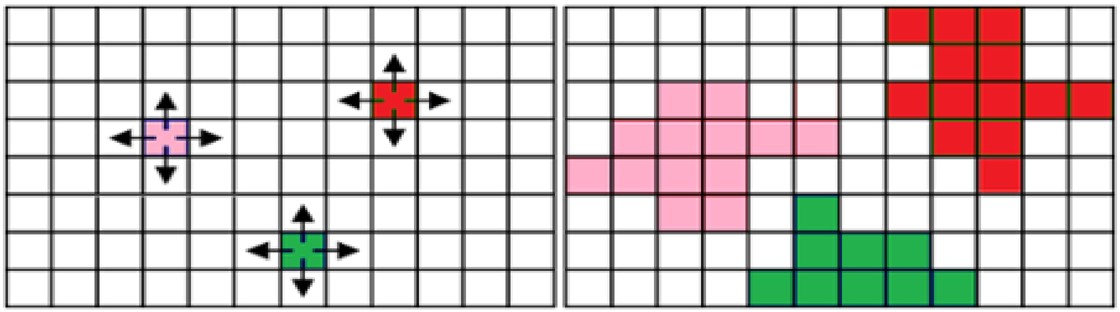


Figure 2.7: Region Growing Segmentation Illustration

Figure 2.8: Region Growing Example

Split and Merge segmentation is complicated, time consuming but give more accurate results compared to the previously discussed techniques. Split and merge techniques takes any image as single region then apply top-down algorithm to define new regions. The process of splitting continue until all the formed regions are uniformed (Li et al., 2015).

2.2.2 Technique-based Segmentation Classification

Based on the utilized technique, image segmentation techniques can be classified into edge linking, clustering, region growing and graph-based. All of these are contextual-based segmentation, except the clustering, which can be both contextual and non-contextual based.

2.2.2.1 Edge-Linking Segmentation

Edge linking technique finds edges, which are clear dissimilarity between the pixels in the image then link the adjacent edges in order to form a boundary that isolate objects from each other. As a contextual segmentation, the edge linking approach operates over the input image and considers the adjacency relationships between the pixels in order to find edges and edges connectivity. Figure 2.9 illustrates an example of edge detection technique (Yogamangalam & Karthikeyan, 2013).

Figure 2.9: Edge Detection Example

2.2.2.2 Region-Growing Segmentation

Region growing, as discussed earlier, is a simple image segmentation method, which is also called as a pixel-based image segmentation since it involves the selection of initial seed points followed by linking pixels to each other based on the adjacency relationships between the pixels (Holz & Behnke, 2014).

2.2.2.3 Clustering-based Segmentation

Clustering technique groups a set of pixels in such a way that objects in the same group (called a cluster) are similar (in some sense or another) to each other and are different to those in other groups "clusters" (Kamdi, 2012).

The problem with the clustering technique is that the generated clusters do not directly equivalent to image regions, as the clustering process is implemented over the histogram or other feature spaces. A CCQ image (color consistency quantization) is an approximate map to the original image, to generate clusters from the image that can be used with mean-shift & k-means clustering algorithms. The quality of clustering depends on both the similarity measure and its ability to discover hidden patterns.

Many clustering algorithms use the center based cluster criterion, so it is very important to find an effective way to select initial centroids in order to influence efficiency of k-means clustering (Bradley & Fayyad, 1998). An approach for clustering-based image segmentation is illustrated in Figure 2.10.

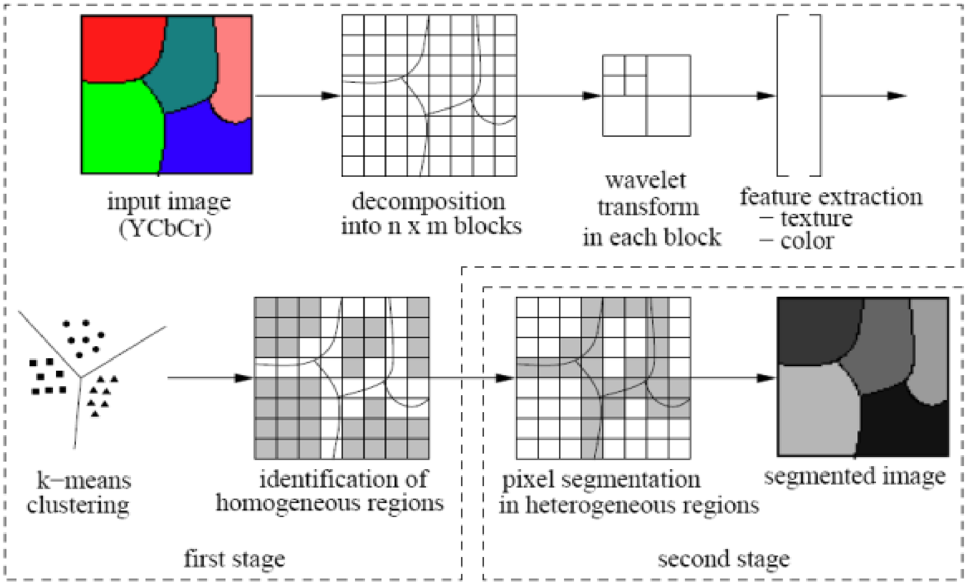


Figure 2.10: An Approach for Clustering-based Image Segmentation (Da Silva & Pedrini, 2011)

2.2.2.4 Graph-based Segmentation

One of the most effective ways to segment colored images is using graph-based technique. In graph-based image segmentation, each pixel of the image is associated with a node on the graph and edges weight is measured by dissimilarity between pixels. Segmentation on the graph, as illustrated in Figure 2.11, is implemented by selecting low weight edges to define a segment and high weight edges to define boundaries between segments or regions (Huang et al., 2012).

The process of creating a graph from an input image is implemented as follows: Pixels are used to create the graph nodes and the intensity of the pixels form the node values. Edges are the boundaries between the pixels, these edges are undirected binary edges. More edges are required in the 8-connected pixels' neighborhood system compared to the 4-connected pixels' neighborhood system. As the graph is generated, there are two techniques that can be used, these are normalized cut and MRFs Graph cut segmentations (Boykov & Jolly, 2001).

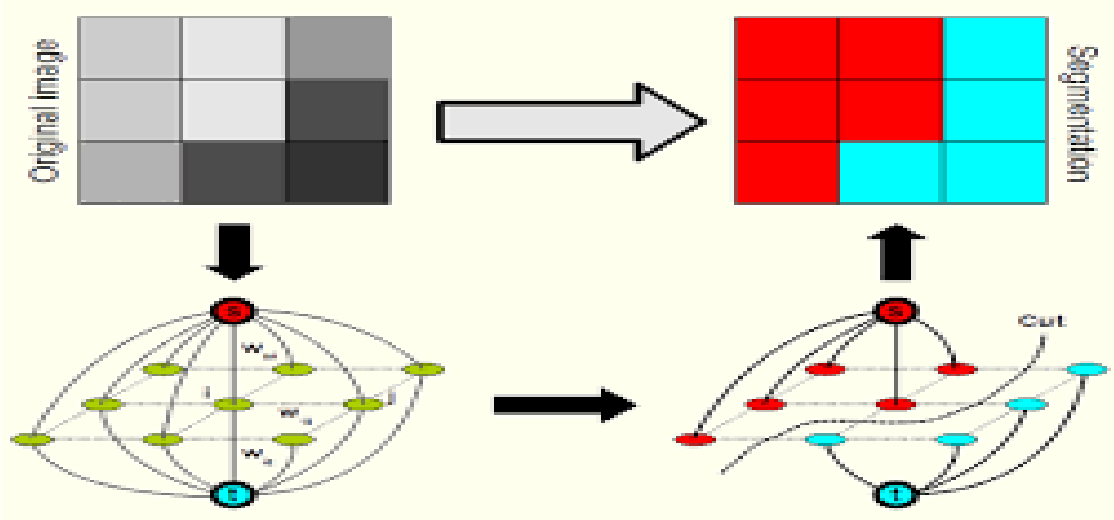
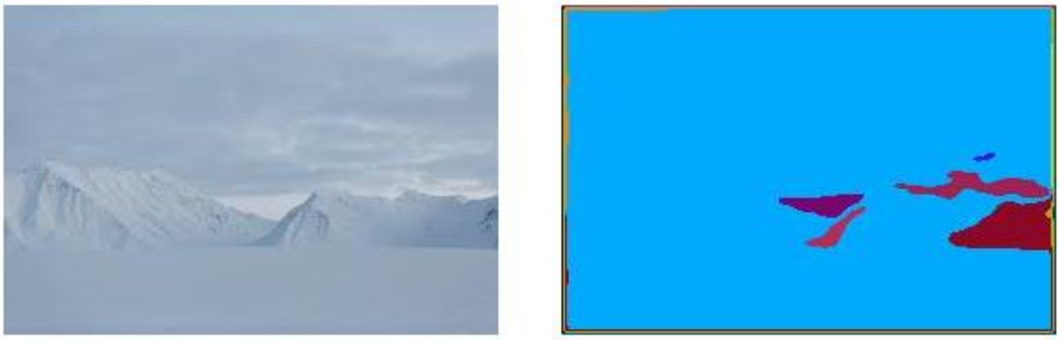


Figure 2.11: Graph-based Image Segmentation

In the normalized cuts segmentation, edges that link pixels of diverge intensities are removed sequentially in-order to form regions. Within the same region, each node should be connected to other nodes by at least a single edge, while among the regions, no pixels are linked. This process formed various regions with connected pixels. Normalized cut technique is not fast nor accurate (Zhang & Hancock, 2008). MRFs Graph cuts segmentation, also called graph-based segmentation, is preferred because of its accuracy and efficiency, although it uses a greedy algorithm. This technique is effective for images with homogeneous colors, but this is not the situation with natural images, or to be more specific images with various color levels (Vu, 2008). The main challenges that faces MRFs technique, which may lead to bad segmentation as illustrated in Figure 2.12, are as follows (Wojnar, 1998).

- Image noise: Noise leads to wrong segmentation, as the noise maybe appeared as region boundary. Smoothing of image to eliminate the effects of noise, may lead to lose these boundaries.
- Various object size: Actual objects may be smaller or larger than the generated regions, small objects may be merged within larger regions.



Original Image Segmented Image

Figure 2.12: Result of Bad Image Segmentation

Generally, graph, based segmentation, the subject matter of this thesis, examines the weights to define the regions according to the intensity differences across the boundary and intensity difference between neighboring pixels within each region. A boundary is defined when there is obvious difference in intensity between the underlying pixels. Minimum or no difference between pixels lead to locate these pixels in the same region, as illustrated in Figure 2.13. With each iteration on the graph, the edges with low values are preserved and the nodes in the same segment are merged. While other edges with high value are deleted and the pixels might be located in different regions, if all the connected edges are removed (Felzenszwalb & Huttenlocher, 2004).

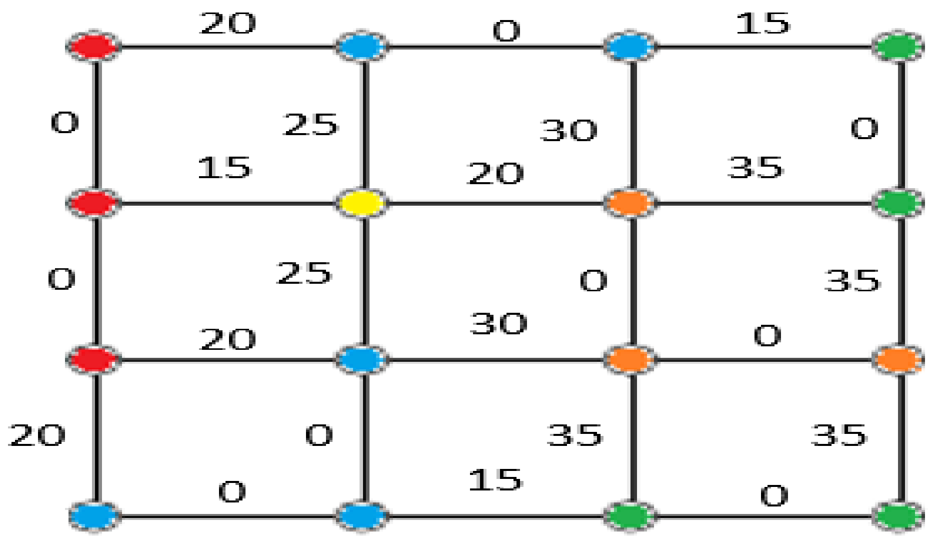


Figure 2.13: Undirected Graph with 4x4 Weighted Edges Image Representation

Graph-based image segmentation technique as represented by Felzenszwalb & Huttenlocher, (2004), is implemented based on the following steps:

1. The image is loaded and converted into undirected graph. Each pixel is represented by a node in the graph and pixels neighborhood is represented by arcs.

2. The weight of each edge in the graph is calculated as the difference in intensity (i.e.: gray scale or color value) between the two adjacent nodes as illustrated in Figure 2.13.

3. The set of edges in the graph are sorted based on its weight in non-decreasing order.

4. Segmentation starts from the first edge in the sorted list. The weight of the edge is compared to the value of the threshold, K. If the weight is less than or equal to K, then, the nodes that are connected by the underlying edge is merged together or union to form a region. Figure 2.14 illustrates and example of testing and merging nodes with threshold value equal to 5.

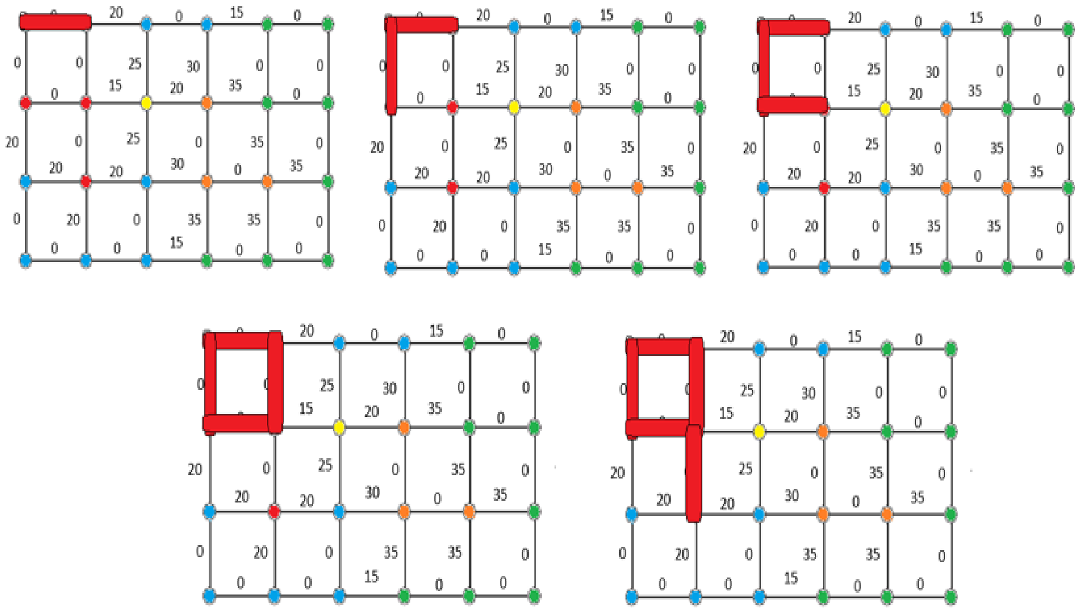


Figure 2.14: Example of Nodes Integration and Region Forming Process

5. As the listed is sorted, all the edges below some threshold are used to form a union between nodes, while all other, at the end of the list ae not used to form regions. In this case, there will be no more union to be implement at this iteration, as illustrated in Figure 2.15.

6. The value of union nodes are calculate by the average of their nodes, and re-calculate the edges that links between different unions as the difference in intensity between the unions' averages.

7. The value of the threshold is calculated as given in Equation 2.1.

$$\textit{New threshold} = \textit{edge weight} + (S_i \, / \, \textit{old threshold}) \qquad (2.1)$$

where S_i, is the average intensity of segment's i.

8. Repeat Steps 3-7 multiple times, until no more union can be implemented anymore.

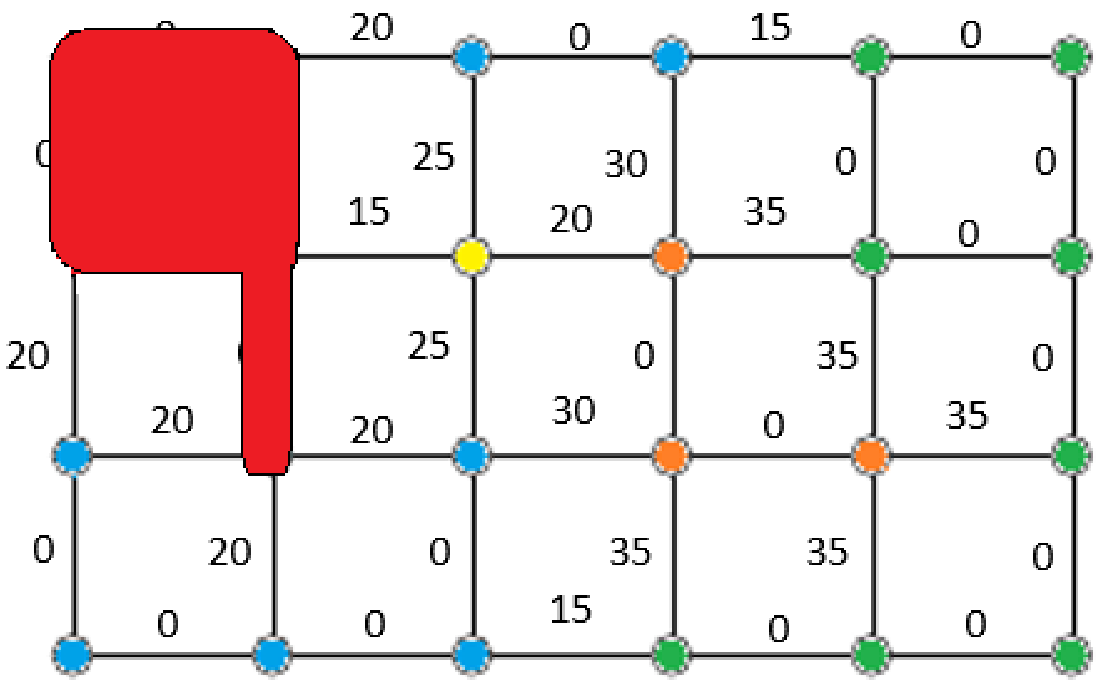


Figure 2.15: Ends of Possible Unions Situation

Based on the described algorithm, a boundary, in the segmented image, exists if the difference between nodes on the sides of the boundary, is larger than the minimum internal differences within the regions on the both sides of the boundary, as given in Equation 2.2 (Martin et al., 2004).

$$\text{Boundary}\ (C_1, C_2) = \begin{cases} \text{True: diff}(C_1, C_2) > \text{Minimum}\ (C_1, C_2) \\ \text{False: else} \end{cases} \tag{2.2}$$

The above steps run in O (n log n) time, where n is the number of pixels in the image. In this thesis, the aim is to start the segmentation process from initial points which will minimizes the number of iteration and reduce the required time.

2.3 Related Work

As mentioned earlier, there is a potential opportunity in obtained an accurate segmentation output using the graph-based image segmentation technique. However, it is necessary to improve graph-based image segmentation accuracy and reduce its time consumption to facilitate the utilization of segmentation in various real-life applications, such as medical imaging, remote sensing and etc. Obviously, a potential opportunity in reducing the process of graph-based image segmentation is exist, which can be implemented by avoid the examination the relationships of connected pixels that are obviously falls into same/different regions. This can be done using automatic initial point selection techniques. Generally, there are many methods and techniques that used graph-based image segmentation. Similarly, many methods were proposed for selecting initial points for clusters. Given below, literature reviews on initial point-based segmentation and extended graph-based segmentation.

2.3.1 Related Work: Initial Point-based Techniques

Initial point has been automatically identified and used for image segmentation, in various techniques, some of these techniques are discussed below.

Nazeer & Sebastina, (2009) proposed an enhanced algorithm to improve the accuracy and efficiency of the K-means (Hartigan & Wong, 1979) clustering algorithm by combining two approaches: First, initial centroid for each cluster is selected automatically by mining the distances, similarities and dissimilarities between the data points in the training set. Second, the K-means algorithms is adjusted to assign data points to the clusters, which is initially created by the initial point. The results of this algorithm show that using initial point for clustering reduce the time complexity without sacrificing the accuracy of the clustering output.

Fahim et al., (2009) proposed an efficient algorithm for clustering, which depend on appropriate initial point selection to allow the k-means algorithm to converge to a better local minimum. Convergence of the clustering was intended to be ends with a better clustering set and using low iteration numbers compared to the original k-means algorithm. Similar to the Nazeer & Sebastina, (2009), the results of this algorithm show that using initial point for clustering reduce the time complexity while enhancing the accuracy of the clustering output.

Shafeeq & Hareesha, (2012) proposed an algorithm for clustering, which depend on initial point selection and variable K-size. The proposed algorithm can be used weather the value of K is known or not. Similar to the Nazeer & Sebastina, (2009) and Fahim et al., (2009), the results of this algorithm show that using initial point for clustering enhance the accuracy of the clustering output but there was no evidence that the number of iterations was reduced.

Islam & Ahmed, (2013) proposed a combined clustering algorithm with initial point selection. The proposed algorithm combined and analyze the effectiveness of K-Means, K-Medoids (Kaufman& Rousseeuw, 1987) and Hierarchical clustering (Corpet, 1988). The results show that initial point-based clustering consume less time but might not enhance the results.

Zheng & Zhang, (2014) proposed two algorithms based on existing clustering algorithms for image segmentation with a way of initial point selection. The first is to generalize Fuzzy C-Means clustering algorithm (Bezdek & Full, 1984) in order to overcome the insufficient robustness to image noise problem. The second was an extension of Hierarchical Fuzzy C-Means clustering algorithm (Dembele & Kastner, 2003) in order to overcome the sensitivity of Euclidean distance to outliners by initial points. The results of these algorithms show improvement on robustness and effective image segmentation.

Various other techniques and algorithms that use initial points were proposed, such as the algorithms proposed by Fayyad & Bradley, (1998), Tishby & Slonim, (2001), Khan & Ahmad, (2004), Meila & Heckerman, (2013), Guha & Mishra, (2016), Goel & Srivastava, (2017). Overall, using initial point with techniques that depends on several iterations to reach good distribution of data points lead to enhance the outcome of the distribution (i.e.: clustering, segmentation, classification and etc.). Moreover, these points reduce the number of iterations required to reach the optimal solution. Table 2.1 summarizes the related work on initial point-based techniques.

Table 2.1: Summary of Initial Point-based Techniques

Author (Year)	Reference Algorithm	Results
Fayyad & Bradley, (1998)	K-Means	Time and Accuracy Enhancement
Tishby & Slonim, (2001)	Markov-based	Accuracy Enhancement
Khan & Ahmad, (2004)	K-Means	Accuracy Enhancement
Nazeer & Sebastina, (2009)	K-Means	Time Enhancement
Fahim et al., (2009)	K-Means	Time and Accuracy Enhancement
Shafeeq & Hareesha, (2012)	K-Means	Accuracy Enhancement
Islam & Ahmed, (2013)	K-Means, K-Medoids and Hierarchical clustering	Time Enhancement
Meila & Heckerman, (2013)	Expectation Maximization (EM)-based	Accuracy Enhancement
Zheng & Zhang, (2014)	Fuzzy C-Means and Hierarchical Clustering	Accuracy Enhancement
Guha & Mishra, (2016)	K-Median	Accuracy Enhancement
Goel & Srivastava, (2017)	K-Means	Accuracy Enhancement

2.3.2 Related Work: Graph-based Segmentation

Graph-based image segmentation technique has been proved to produce a good output segmented images with the ability to be further enhanced and integrated with other techniques. Subsequently, various techniques have been proposed to enhance and improve this technique accordingly.

Felzenszwalb & Huttenlocher, (2004) proposed a segmentation algorithm that implements greedy decisions, but, yet it satisfies pre-determined properties, the algorithm is applied after measuring the boundaries between regions on a graph-based representation. First, using local neighborhoods system, the input image is used to construct a graph, then, the segmentation is implemented by union and disunion nodes in the graph to form regions and boundaries. The proposed segmentation was tested with both, real image and synthetic image. The results show that that proposed algorithm has the ability to preserve detail in low-variability image regions while ignoring detail in high-variability regions, which distinguish this algorithm from other existing algorithms for image segmentation.

Ren et al., (2006) proposed an image segmentation algorithm that uses the concept of defining local shapes by logistic classifier and they find that local shape model achieves an accuracy of 64% in predicting the correct figural assignment. The local shape construction uses a method to compute its edge map using Probability of Boundary (PB) operator. The local shape is used to train a classifier on the intended shapes. A linear classifier, which is trained in the previous stage, is then used to predict the figure/ground label at each image location. While, this algorithm does not use graph directly, it uses the concepts of nodes and arcs in describing the local shape model.

Wassenberg et al., (2009) proposed a Minimum Spanning Tree (MSP)-based algorithm with a novel graph-cutting heuristic. The proposed algorithm converts the input image into a single graph, then the well-known MSP algorithm is used to find the spanning tree that describes regions and links between regions (i.e.: boundaries). Moreover, in order to reduce the high computational cost for MSP algorithm, the proposed segmentation was executed in parallel manner. However, the algorithm still consumes comparably high computational resources, which makes it inapplicable in real-life applications. The results show that the algorithm produces efficient results on standard images.

Grundmann et al., (2010) proposed an efficient and scalable technique for spatial-temporal segmentation of long video sequences. The proposed work was built using hierarchical graph-based algorithm, which generates high quality segmentations. The proposed algorithm used a clip-based processing that divides the video into overlapping clips in time, and segments them successively while enforcing consistency to improve the scalability of the technique. The proposed algorithm was implemented using parallel out of-core model that can process much larger volumes compared to an in-core model. The results show that the algorithm produces efficient results on large volume images captured from video cameras.

Pawn et al., (2010) proposed an algorithm that define the energy function of regions resulted from the segmentation process. The regions are linked by their boundaries in a graph-like manner. The resulted segments are modelled and then merged to form larger regions based on the adjacency relationships. The proposed algorithm was proven to give a better understanding of a given scene embodied in the resulted energy function. Moreover, it was proven that the segments obtained from bottom-up approaches gives more reliable demonstration of the objects in the scene and can be used to extract reliable feature representation of the underlying scene.

Mobahi et al., (2010) proposed a novel algorithm for natural image segmentation using minimum description length. The proposed algorithm encodes the length between the regions as a function of texture in a graph structure. Then, the segmentation algorithm is implemented over the graph with the goal to reduce the description length within the region, by having coherent structure and increase the description length among the regions by construct incoherent structure between these regions. The results show that the proposed algorithm produces an effective code for encoding boundaries of image and achieve the-state-of-the-art segmentation results.

Kumar et al., (2010) proposed a segmentation algorithm that implements both bottom-up and top-down approaches over a graph corresponding to the input image. The findings of the proposed algorithm can be summarized as follows, 1) Obtain the pose of the input image without human interfered. 2) Obtain the shape and pose of the objects using an object category model. Another great achievement is their framework ability to process large intra class shape, appearance, and spatial variation.

Hickson et al., (2014) proposed a scalable algorithm for videos segmentation, using multi-stage processing technique. The proposed technique can process videos regardless of the length by a moving window over several time windows. The proposed technique constructs a graph-based from the input scene and implements regions merging using minimum spanning tree. Overall, this technique can be summarized in four stages, these are: First: frame segmentation over n-consecutive frames (the experiment was performed using n = 8). Second: image over segmentation with the reference of color and motion with respect to previous segmentation. Third: hierarchical segmentation with the reference of region's histogram. Fourth: bipartite graph matching frames using frame overlapping to enforce the consistency of region identities over time.

Yang et al., (2015) proposed a segmentation technique for 3D volumes, using graph and initial point selection. The proposed technique consists of two levels: First, super pixel segmentation process, which uses geometry to allocate initial points and construct regions in the image and produces over segmented map. The over segmented image is then represented as a graph of nodes that encodes regions and arcs that encode dissimilarity/similarity between regions. Second: A graph-based merging with a K-means like clustering is used to merge super pixels into larger regions. The proposed approach uses RGBD proximity, texture similarity, and boundary continuity to decide about the merging of neighboring super pixels. The experimental results show that the proposed technique over-performed several state-of-the-art algorithms for image segmentation.

Burdescu & Slabu, (2015) proposed two algorithms for image segmentation using graphs. The first algorithm creates a hexagonal structure on the image and hexagons on graph. Then, using a modified form of the Kruskal's algorithm, a minimum spanning tree algorithm to generate a set of color-based region models. Then, the determined regions are used to extract color and geometric features from the resulted segments. The second method is used to segment spatial images, using virtual tree-hexagonal network superposed over the initial image voxels. Compared to the first algorithm, this method reduces the execution time and maintain the accuracy of the output results.

Other algorithms for graph-based image and video segmentation were proposed in the literature. The goal of the graph-based segmentation is to use the graph to address the adjacency relationships between neighborhood pixels, super pixels and regions. These adjacency relationships are used to guide the merging process in bottom-up segmentation approaches or to guide the splitting process in top-down segmentation approaches. Overall, graph-based segmentation techniques were proven to provide good results, better understanding of the scene and efficient feature modeling. Table 2.2 summarizes the related work on graph-based image segmentation techniques.

2.4 Summary

In this chapter, the existing literature on image segmentation and the technique that uses initial points to direct the processing of samples and images were discussed. There are various image segmentation categories, such as thresholding, region growing, clustering and graph-based segmentation. Graph-based technique was proved to provide good results, better understanding of the scene and efficient feature modeling, yet it consumes time. Subsequently, initial point-based processing was used in the literature with segmentation and other segmentation-like applications, such as clustering in order to reduce the number of required iterations without affecting the overall mechanism that is built over a graph representation. Table 2.3 summarizes the techniques, which was addressed in this chapter.

Table 2.2: Summary of Graph-based Segmentation Techniques

Author (Year)	Reference Algorithm	Results
Felzenszwalb & Huttenlocher, (2004)	Bottom-Up	Accuracy Enhancement
Ren et al., (2006).	Bottom-Up	Accuracy Enhancement
Wassenberg et al., (2009)	Bottom-Up	Accuracy Enhancement
Grundmann et al., (2010)	Bottom-Up	Accuracy Enhancement
Pawn et al., (2010)	Bottom-Up	Accuracy Enhancement
Mobahi et al., (2010)	Bottom-Up	Accuracy Enhancement
Kumar et al., (2010)	Bottom-Up	Shape-Recognition
Hickson et al., (2014)	Bottom-Up	Time Enhancement
Yang et al., (2015)	Bottom-Up	Accuracy Enhancement
Burdescu & Slabu, (2015)	Bottom-Up % Top-Down	Time Enhancement

Table 2.3: Summary of the Literature

Author (Year)	Graph-Based	Initial-Point based	Reduce Complexity
Felzenszwalb & Huttenlocher, (2004)	√	X	X
Ren et al., (2006).	√	X	X
Wassenberg et al., (2009)	√	X	X
Grundmann et al., (2010)	√	X	X
Pawn et al., (2010)	√	X	X
Mobahi et al., (2010)	√	X	X
Kumar et al., (2010)	√	X	X
Hickson et al., (2014)	√	X	√
Yang et al., (2015)	√	X	X
Burdescu & Slabu, (2015)	√	√	√
Fayyad & Bradley, (1998)	X	√	√
Tishby & Slonim, (2001)	X	√	X
Khan & Ahmad, (2004)	X	√	X
Nazeer & Sebastina, (2009)	X	√	√
Fahim et al., (2009)	X	√	√
Shafeeq & Hareesha, (2012)	X	√	X
Islam & Ahmed, (2013)	X	√	√
Meila & Heckerman, (2013)	X	√	X
Zheng & Zhang, (2014)	X	√	X
Guha & Mishra, (2016)	X	√	X
Goel & Srivastava, (2017)	X	√	X

As noted, initial-point automatic selection is mostly associated with the time and complexity reduction. Subsequently, it can be used to enhance the time complexity of the graph-based segmentation technique. Moreover, the existing algorithms for graph-based were leak the motivation to reduce time, which make most of them of high complexity. The graph-based technique, which was proposed by Felzenszwalb & Huttenlocher, (2004) is accurate and efficient, yet it can be further enhanced accordingly.

Chapter Three

Methodology

In this chapter, an extension to the graph-based image segmentation technique is proposed in order to reduce the time consumption while preserve the segmentation quality. The organization of this chapter is as follows: Section 3.1 is an introduction. Section 3.2 presents the overall framework of the proposed work. Section 3.3 discusses the process of image smoothing. Section 3.4explains the way by which the image is converted into a graph. Section 3.5 proposed an approach for initial point selection. Section 3.6 discusses the proposed algorithm for image segmentation based on initial points. Section 3.7 illustrates an example. Finally, a summary is given in Section 3.8.

3.1 Introduction

The existing graph-based image segmentation technique, which was proposed by Felzenszwalb & Huttenlocher, (2004) uses greedy algorithm that consumes nearly linear time. The aim of the proposed work is to reduce the time consumption by defining initial points in the image and applies the algorithm depending on these points. Initial point is a pixel with similar feature (color for example) to the pixels around it that makes is the core of a homogenous region. Initial point selection has been reported in the literature with region-growing segmentation techniques, data clustering and clustering-based segmentation. Subsequently, using initial point to form homogenous regions was tested and verified. In this chapter, initial points are automatically identified and employed in graph-based image segmentation.

3.2 The Proposed Framework

The proposed framework, as illustrated in Figure 3.1 is composed of various steps that started with the input image and produces a segmented image. As mentioned, the goal of the given framework is to minimize the required time without effecting the efficiency of the segmentation results by decreasing the number of iterations needed. The process that are included in the proposed framework are: image smoothing, image-to-graph conversion, initial points selection and segmentation, those steps are discussed in the following sections.

The input image is first smoothed and the resulted smoothed image is then used to construct a graph representation of nodes and arcs. The smoothed image is also used as input for initial point selection, by which initial points are identified in order to guide the segmentation process. Finally, the segmentation process uses the initial point and the graph representation in order to find boundaries and forms regions over the original input and the final output is produced accordingly.

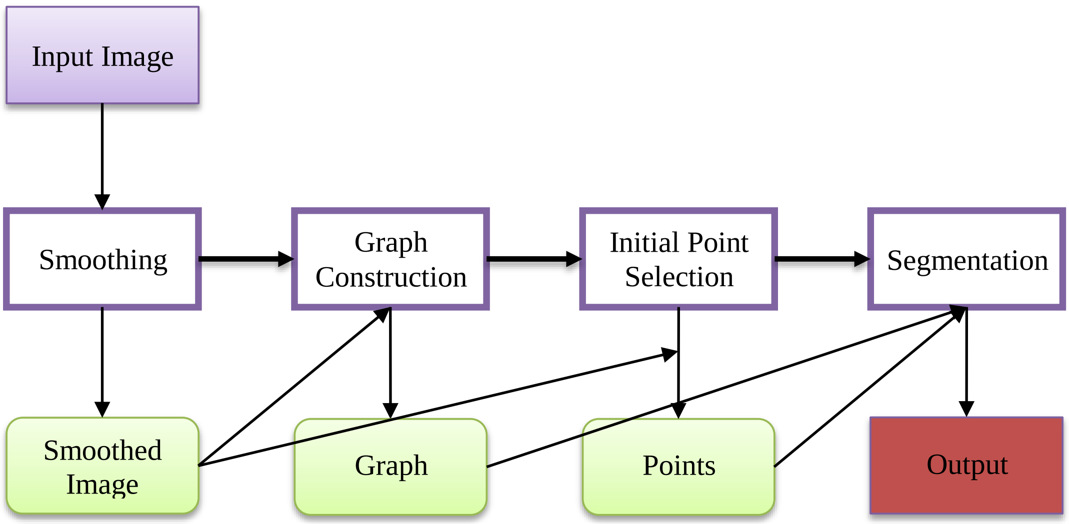


Figure 3.1: Flow Chart of the Proposed Framework

3.3 Image Smoothing

Images captured by typical digital cameras, which are called natural images, are subject to variety of disturbing effects, such as noise and illumination. Subsequently, smoothing are required in order to enhance the quality of the images, which will enhance the output of the segmentation process. Another important benefit of smoothing and image is having good results regardless to the image resolution. There are variety of purposes for image smoothing, such as noise removal and gesture marking, each of these purposes are implemented by variety of methods.

In the proposed work, the smoothing is implemented in several stages, as illustrated in Figure 3.2. The input image is first decomposed into their color channels, red, green and blue and the noise is removed based on comparing each pixel channel with its neighborhood. Then, the resulted images after decomposition and noise removal is processed by gesture. Finally, the results images are combined back into a single smoothed image. Figure 3.3 illustrates an example of image smoothing as implemented in the proposed work.

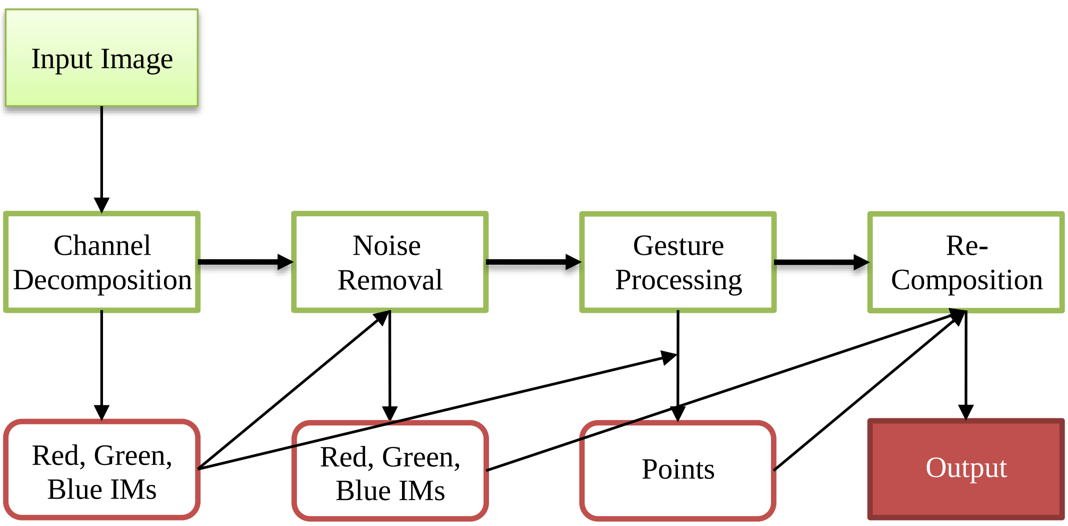


Figure 3.2: Flow Chart of the Smoothing Stage

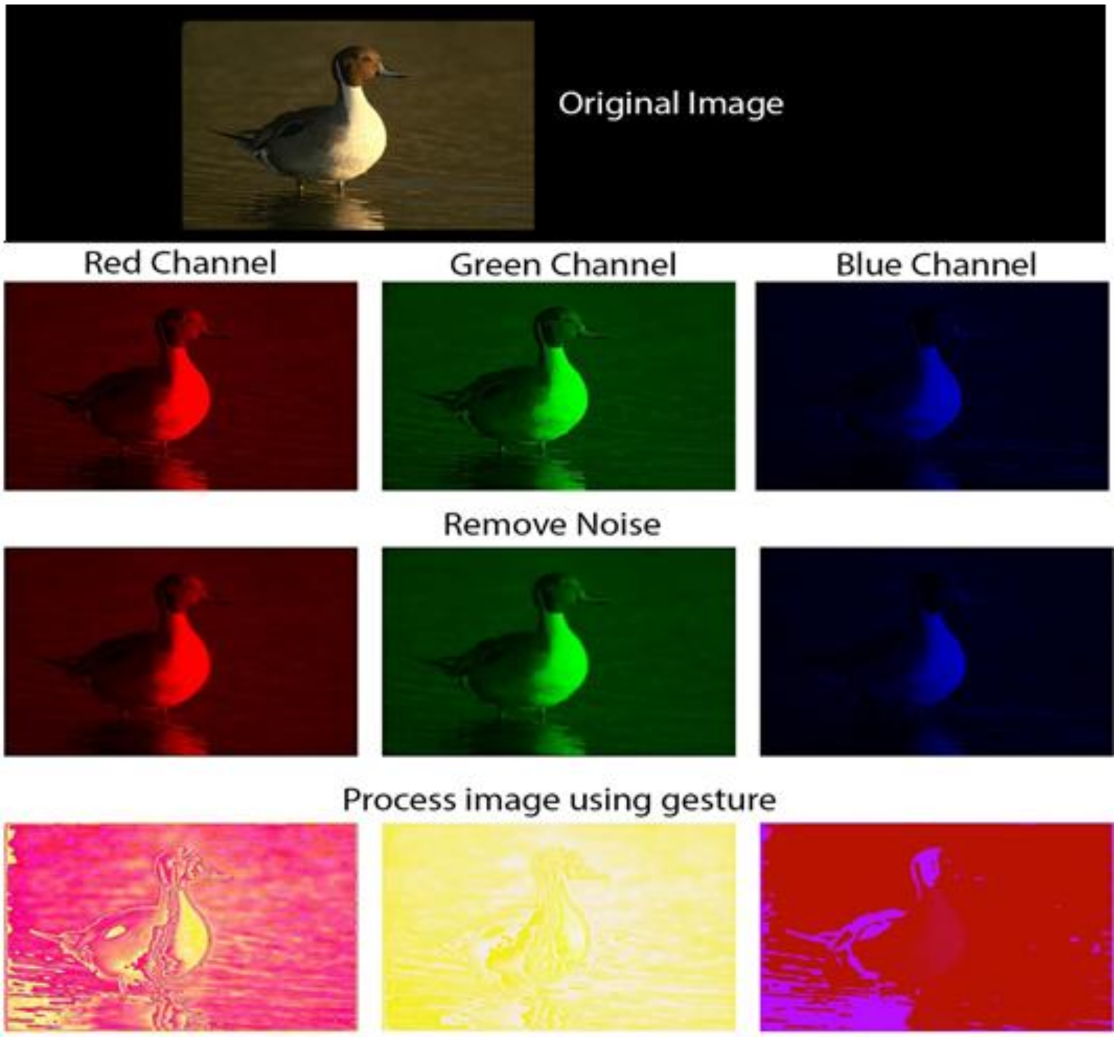


Figure 3.3: Example of the Smoothing Stage Intermediate Outputs

3.4 Image to Graph Conversion

The first step in the proposed technique, as similar to Felzenszwalb & Huttenlocher, (2004), is to read the input image and convert the image into an undirected graph. Pixels in the image are processed iteratively and each pixel is used to create a graph node. Besides, each node, except for the pixels at the border, is linked with four or eight arcs, depends on the neighborhood system utilized. The weights at the established arcs in the graph are calculated as the difference in color, or any other image feature utilized, as illustrated in Figure 3.4. Algorithm 3.1 gives the process of convert image to graph.

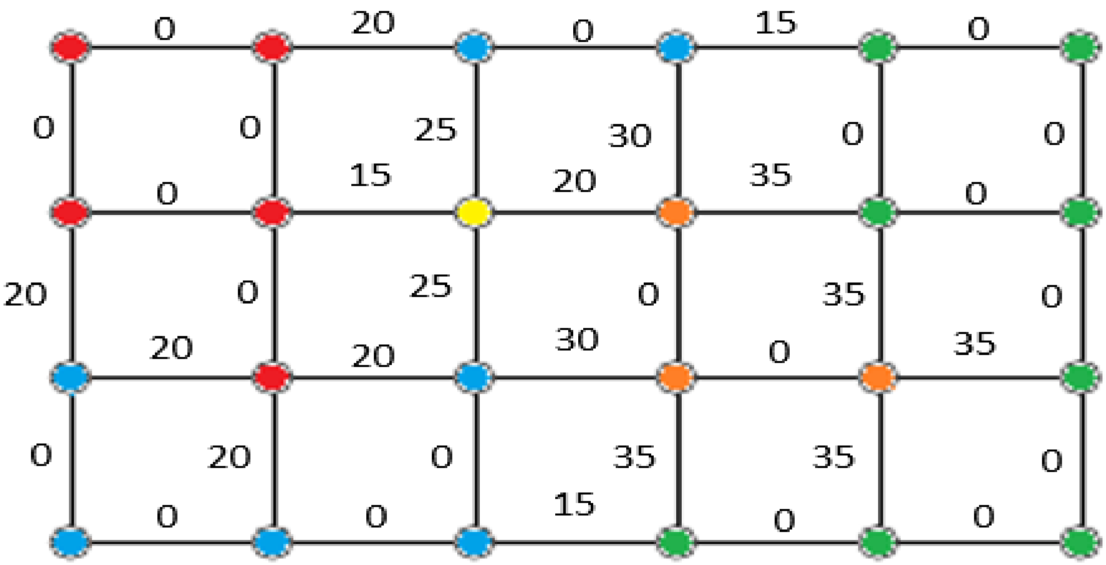


Figure 3.4: Graph Representation of an Image

Algorithm 3.1: Image to Graph Conversion

1. Input Image I(x:0-n-1,y:0-m-1)
2. Create Graph G (Empty Set of Nodes)
3. Foreach Pixel P(i,j) in I, do
4. Create Node n_t : n_t = P(i,j), iff not in G
5. Foreach Neighborhood P*(i*,j*) to P, do
6. Create Node n_u*, if n_u* not in G
7. Create arc $a_v(n_t,n_u*)$
8. Calculate arcWight= |Intensity (n)- Intensity (n*)|
9. EndFor
10. Add n_t, n_u *, a_v(n,n*) to G
11. EndFor
12. Output: G

Line 1 represents the input image as a matrix pf n-rows and m-columns. In line 2, an empty graph without nodes nor arcs are created. Line 3 to line 11 address the actual graph construction process. Line 4 creates a node for every pixel in the input image, if and only if that pixel does not exist in G which occurs with the first pixel only, as the nodes with other pixels will be created in line 6. Then, in line 5 to line 9, the neighborhood pixels are processed. Line 6 creates a node for every neighborhood pixel, if and only if that pixel does not exist in G. Arcs that are connecting nodes and its associated weights are calculated in line 7 and line 8, respectively. Finally, the created nodes and arcs are added to the graph in line 10.

3.5 Initial Points Selection

Initial points are non-boundary points in an image, which are identified based on the difference between two neighborhood pixels in a comparison with predefined threshold. The initial point selection is iteratively process the input points in the image and among a sub-set of identical or similar points (i.e.: data points are said to be similar if and only if the difference in color, or any utilized feature, is below some threshold), a single point is preserved. This process is repeated, until the number of data points reach some pre-defined number of points or until the differences between the data points are above some threshold. The process of initial point selection is illustrated in Figure 3.5, and are described in the following.

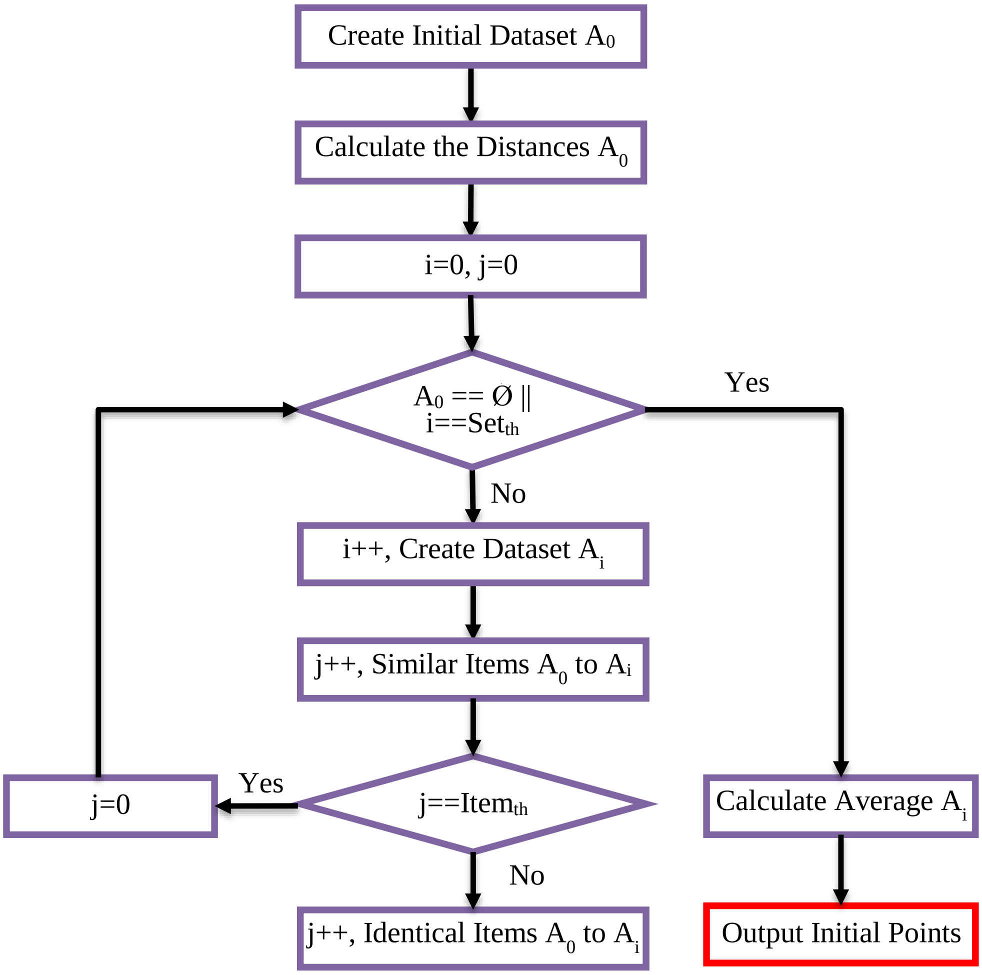


Figure 3.5: Flow Chart of Initial Points Selection

<u>Initial Points Selection Steps:</u>

Step 1: Create an initial dataset (A_0), which contains all the pixels in the image.

Step 2: The distance between each pair of neighborhood points in the data set (A_0) is calculated.

Step 3: Select the closest pair of points in the set (A_0) and move them to a new dataset (A_1), while deleting them from the initial dataset (A_0).

Step 4: Move all the points in (A_0) that are similar to the points in (A_1) iteratively, until the number of data points in A_1 reach a threshold.

Step 5: Select the closest pair of points in the remaining items of the set (A_0) and move them to a new dataset (A_2), while deleting them from the initial dataset (A_0).

Step 6: Move all the points in (A_0) that are similar to the points in (A_2) iteratively, until the number of data points in A_1 reach a threshold.

Step 7: Repeat step 5 and step 6 to create the set Ai, where *i*, is a pre-defined number or until no more data points left in A_0.

Step 8: Calculate the value of the initial point as the points which hold the average value in each dataset A_i.

Subsequently, initial point selection ensures that points that is intermediate among a group of similar pixels are selected. Depends on initial point selection, regions can be formed around these points and boundaries can be identified accordingly. Figure 3.6 illustrates example of boundary identification based on initial point selection, while Figure 3.7 illustrates example of segment identification based on initial point selection.

Figure 3.6: Initial Points based Boundary Identification

Figure 3.7: Initial Points based Regions Identification

3.6 Image Segmentation

Using initial point selection, graph segmentation is implemented by merging pixels to the initial points based on the similarity/difference between the pixels. Yet, initial points are not allowed to be merged into a segment. Subsequently, a more consistent region forming is implemented in the proposed technique compared to the original graph-based segmentation, which allows all the pixels. This consistency reduced the merging-splitting process and produce more accurate results having that the initial points are consistent with each other.

The proposed segmentation process is illustrated in Figure 3.8. The segmentation process depends on using minimum spanning tree to find the segment in the image with extra condition that prevent the process from linking two initial points to each other using high cost plenty of such action. As such, each initial point is used to form a segment in the produced output segmented image. Algorithm 3.2 gives the process of initial point-based graph-based image segmentation.

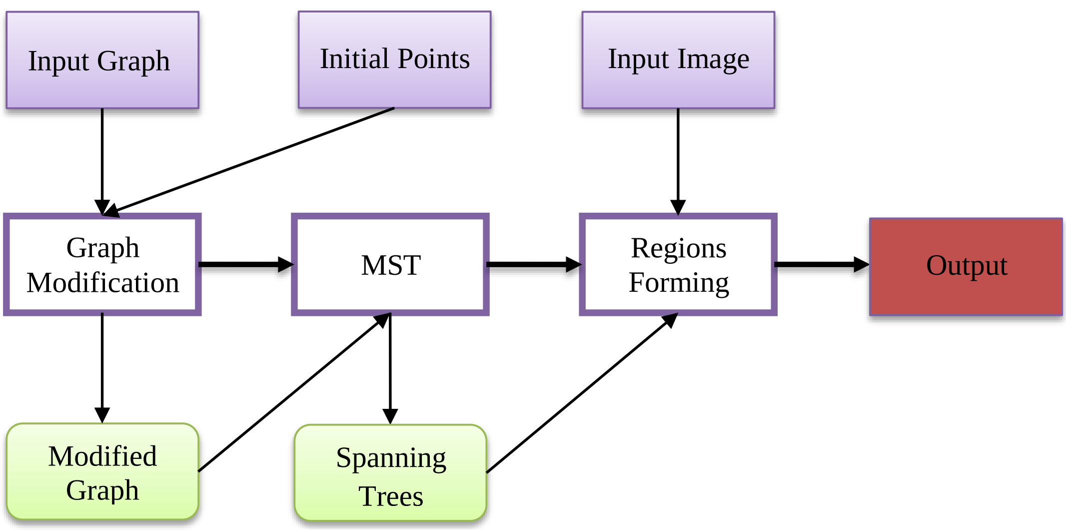


Figure 3.8: Flow Chart of Proposed Segmentation

Algorithm 3.2: Graph Segmentation with Initial Points

1. Input Image I, Graph G, Initial Point S
2. G' = Weight-Adjustment (G,S)
3. ST = MST (G')
4. Segmented-Image = Form(ST, I)
5. Output Segmented-Image

Line 1 represents the input, created graph and the set of initial points. In line 2, the input graph is modified, such as the links between the initial points are removed, by tracing the highest weights among the chain of connected edges and remove them accordingly. Line 3 implement MSP and produce the output trees, which is converted into regions in line 4.

In order to produce images that identifies the boundaries, all the pixels that have no significant differences with any of its neighbors (i.e.: the pixels with remaining connecting arcs/edges) are given one identical color, while the pixels that have significant differences with any of its neighbors (i.e.: the pixels with few or no connecting arcs/edges) are given different color to identify the boundaries in the image. For region identification output, each set of pixels that have no significant differences with each other (i.e.: the pixels that connected to the one point) are given one color.

3.7 Example

In order to illustrate the segmentation process, an example over a given graph is illustrated. Consider the graph that was given in Figure 3.4. The initial points are identified as a pair of identical nodes in the graph, as illustrated in Figure 3.9. As noted, among the graph nodes, the initial points are those linked to another with arcs of weights equal to zero. By other means, initial points are those identical to other nodes, in intensity, within the graph. In the given example, five initial points distributed over the input graph are identified. Apply the algorithm starting from each initial point, using MST and merging nodes will result in several regions, as shown in Figure 3.10.

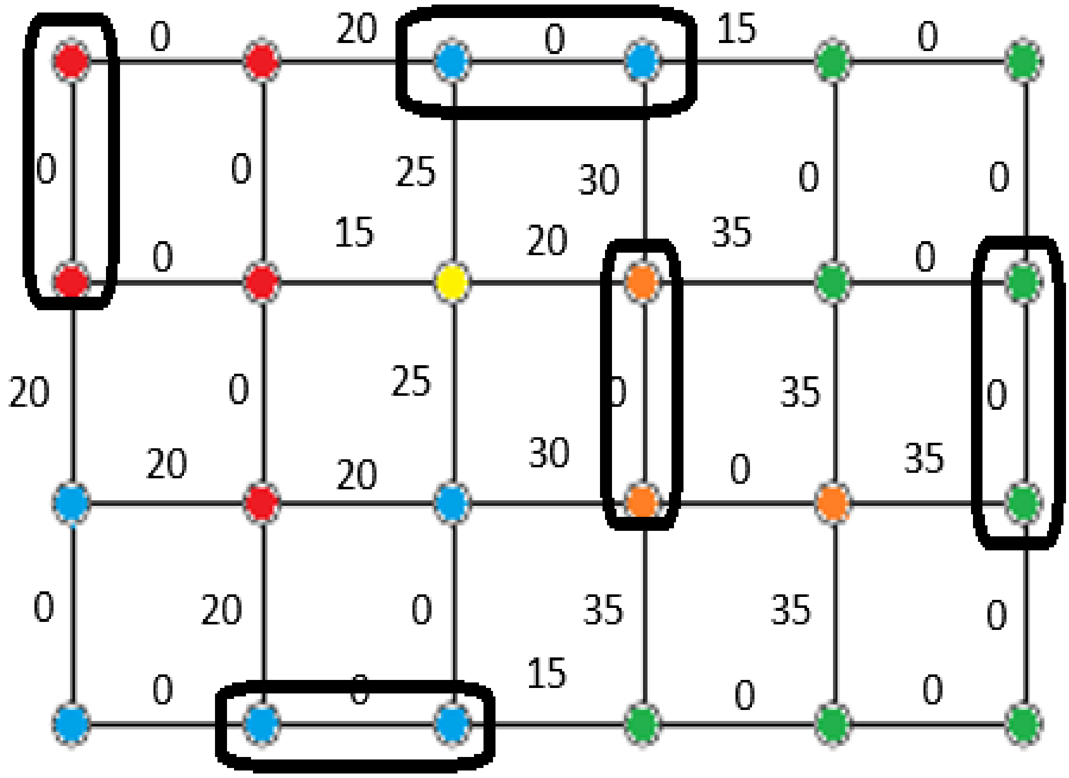


Figure 3.9: Example of Initial Points Selection

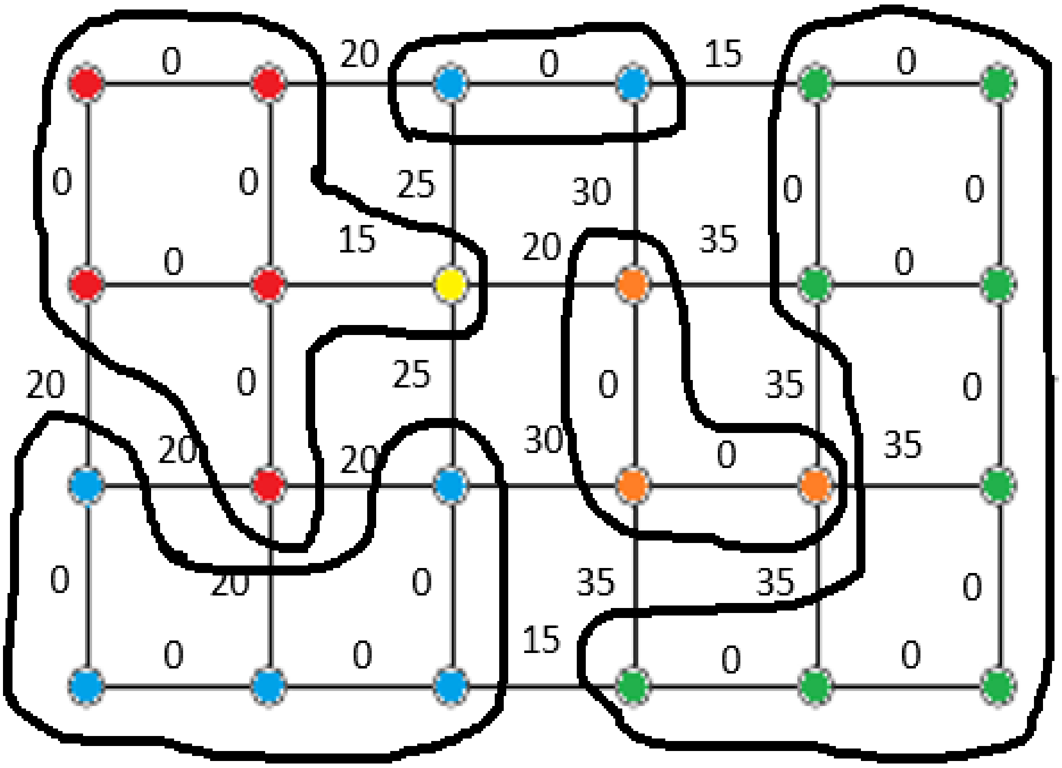


Figure 3.10: Example of Segmented Graph-based Image

3.8 Summary

In this chapter, graph-based image segmentation is implemented based on initial point selection. The proposed technique merge pixels to the initial points based on the similarity/difference between the pixels. Yet, initial points are not allowed to be merged into a segment. First, the input image is smoothed. Then, a graph is created from the input image and the initial points are selected from the input image, as well. According to the initial points, the input graph is modified, such as the links between the initial points are removed. Finally, MSP is implemented and the output trees is used to generate the segmented image.

Chapter Four
Experimental Results and Analysis

In this chapter, the efficiency of the proposed extension is evaluated based on a set of images acquired from an image dataset. The proposed technique is compared with the original graph-based image segmentation (Felzenszwalb & Huttenlocher, 2004). This chapter is organized as follows: Section 4.1 is an introduction. Section 4.2 presents the implementation details. Section 4.3 gives details of the image dataset utilized in the experiments. Section 4.4 discusses the time and accuracy of the proposed technique in comparison with the original technique. Finally, a summary is given in Section 4.5.

4.1 Introduction

In order to validate and evaluate an image segmentation algorithm, a dataset of number of images with its ground truth, which describes, by any means, the correct segmentation of the image, is used. The algorithm, to be evaluated, should be implemented and the output is compared to the ground truth. An accuracy measure, based on the differences between the produce output and the ground truth is calculated as an output for the evaluation process.

4.2 Implementation

The proposed segmentation technique was implemented using Java in NetBeans Integrated Development Environment (IDE) version 8.1. The implementation was conducted with reference to the code provided by Felzenszwalb & Huttenlocher, (2004). The original code provided by Felzenszwalb & Huttenlocher, (2004), was built using *C++*. This code was re-implemented and modified using Java.

The implementation of the proposed technique is implemented in two stages, the second step can be region-based or boundary-based, which result in two different outputs. The first stage, is implemented as follows:

1. Divide the input image into three images each of a single-color channel, for each pixel in the original image, each color value is placed in different image while the other two are set to zero.
2. Smooth each generated image using tow techniques, to discard any unnecessary high difference in similar colors.
3. Build undirected graph of the image, where each edge weight equals to the difference between the two nodes of that edge.
4. Define initial points by combining the neighborhood pixels with minor weight difference.
5. Implement segmentation.

The second stage consists of defining the output segmented images according to one of two methods, produce boundaries image or produce color segmented image. To produce boundary image, pixels within any region is given a single color, while pixels between regions are given different color. While to produce a region image, pixels within each region is given different color.

4.3 Parameters Setting

There are two parameters that are identified by the user, these are K and minSize. The variable K, is a constant represents threshold used for the edge weight, where any edge weight value higher than K represent a boundary. The variable minSize, is a constant that represents the minimum number of pixels to be included in any segment. In the graph-based segmentation, the number of generated segments can be minimized by using high values for Kand minSize. Figure 4.1 illustrates an example of row image and two segmentation results, the first one using low values for K and minSize, which result in segmenting most of the image details, while the second one using high values for K and minSize, which result in segmenting the main object in the image.

Generally, the values of the involved parameters are determined based on the nature of the images to be segmented. There is a need to use different values with indoor images compare to outdoor images. Moreover, medical images totally different from natural images, subsequently, the values of parameters K and minSize, needs to be determined carefully for each type of images.

Common approach for parameter settings is try different values with randomly selected images in the dataset and choose the best values. These values are then used with all the images in the dataset.

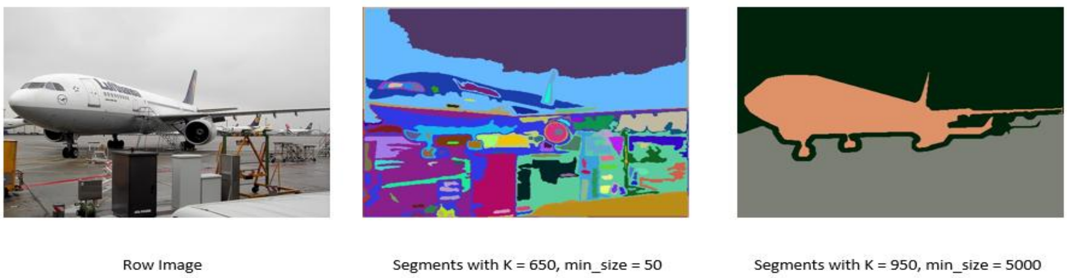


Figure 4.1: Image segmentation using Different Values for K and minSize

4.4 Datasets

To evaluate the efficiency of the algorithm, a set of images with various resolutions, in order to evaluate the time consumption, are used. Samples of these images are given in Figure 4.2. These images are colored images with 256 levels in each channel (RGB) encoded in JPG format. Moreover, second dataset, which is provided by Pascal's challenge with thousands of images and their associated ground truth, are used. Pascal challenge provides standardized image data sets for object class recognition, a common set of tools for accessing the data sets and annotations and enables evaluation and comparison of different methods. Example of Pascal's images are given in Figure 4.3.

Figure 4.2: Example Images from the 1st Dataset

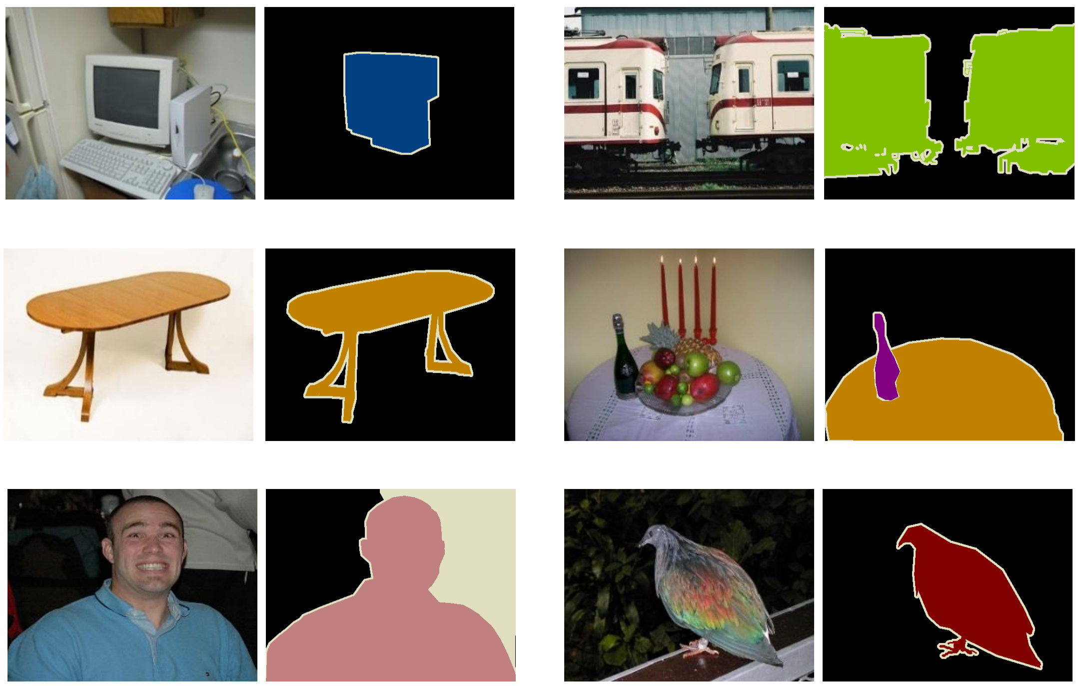


Figure 4.3: Example Images from Pascal Dataset

4.5 Results and Comparison

The experiments are conducted for the proposed technique and the compared technique over the image datasets, which were discussed earlier. Sample of the produced outputs for the first dataset, using the original technique, the proposed technique to produce boundary images and the proposed technique to produce region images are given in Figure 4.4, Figure 4.5 and Figure 4.6, respectively. Sample of the produced outputs for pascal dataset, using the original technique and the proposed technique are given in Figure 4.7 and Figure 4.8, respectively. Time and accuracy of the results are compared in the following sub-sections.

Figure 4.4: Result of Segmentation using the Original Technique on the 1st Dataset

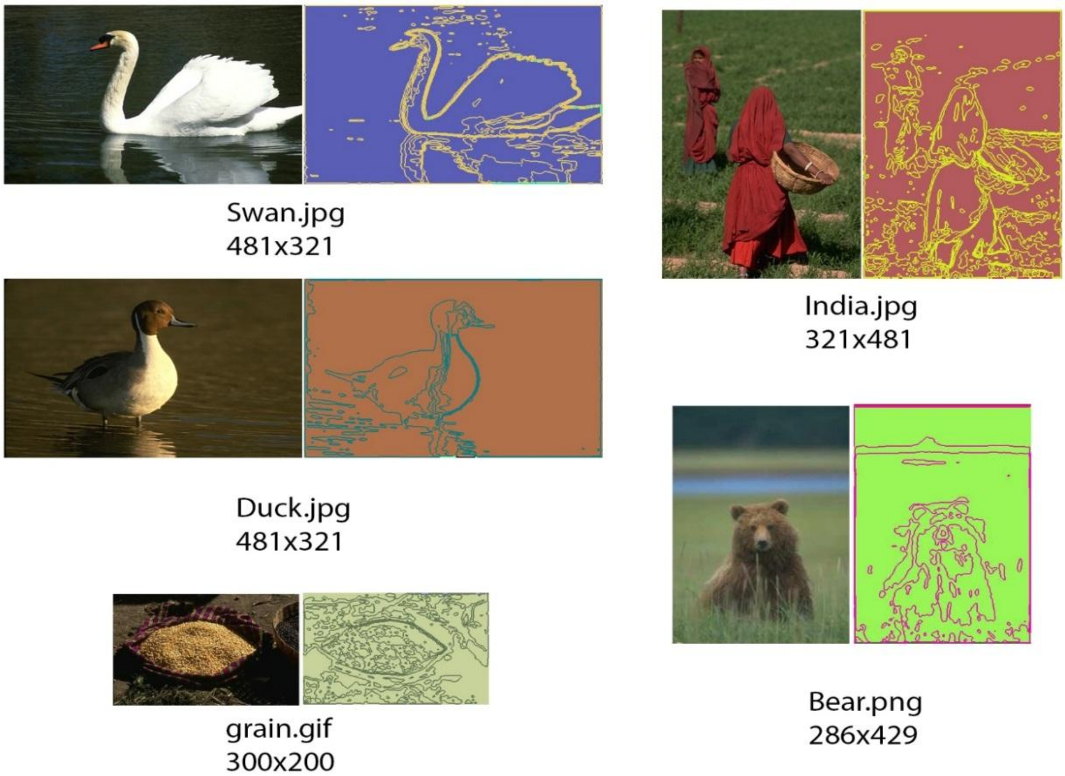


Figure 4.5: Result of Segmentation using the Proposed Technique-Boundary Images on the 1st Dataset

Figure 4.6: Result of Segmentation using the Proposed Technique-Region Images on the 1st Dataset

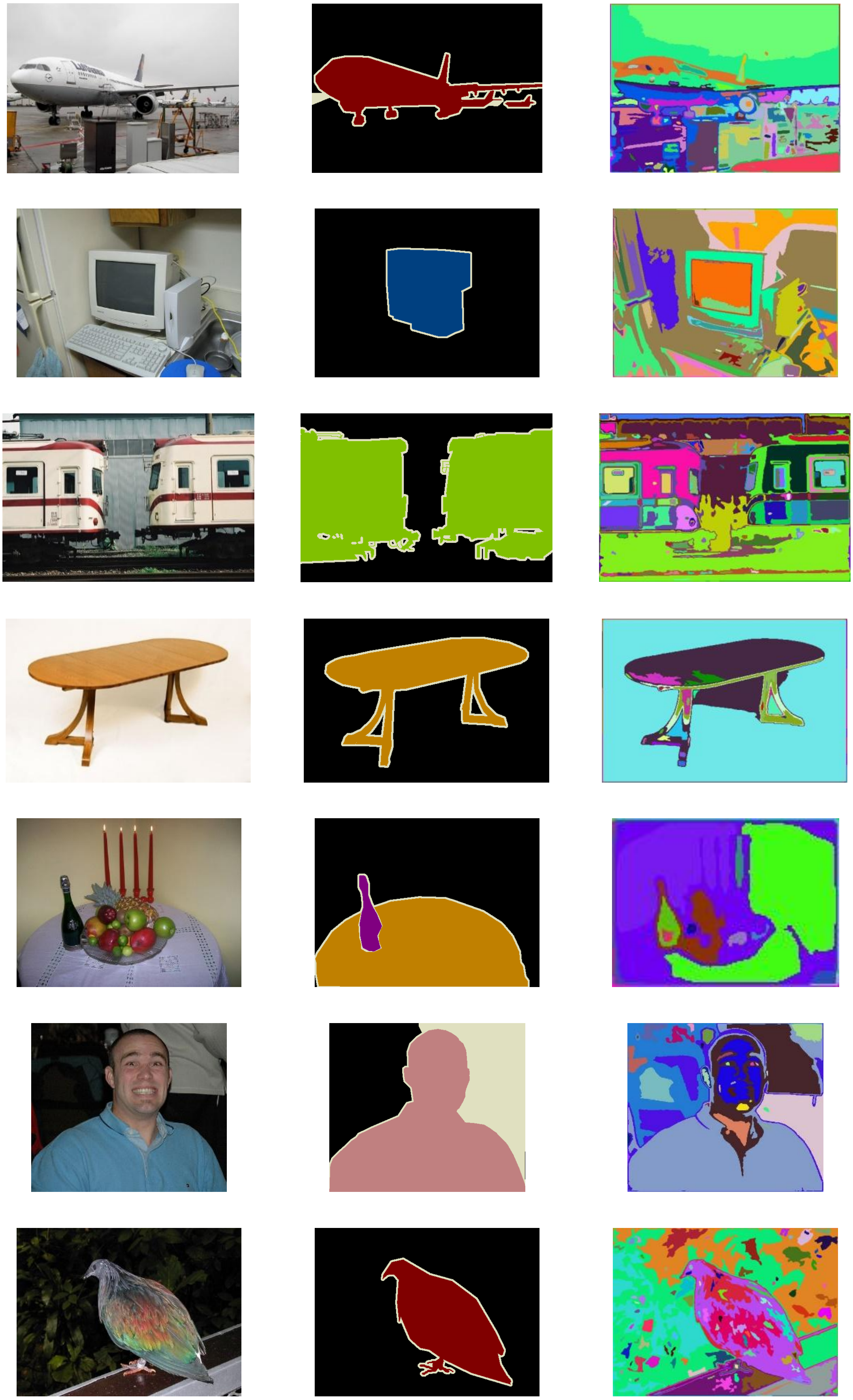


Figure 4.7: Result of Segmentation using the Original Technique on Pascal Dataset, Left: Original Image, Middle: Ground Truth, Right: Output

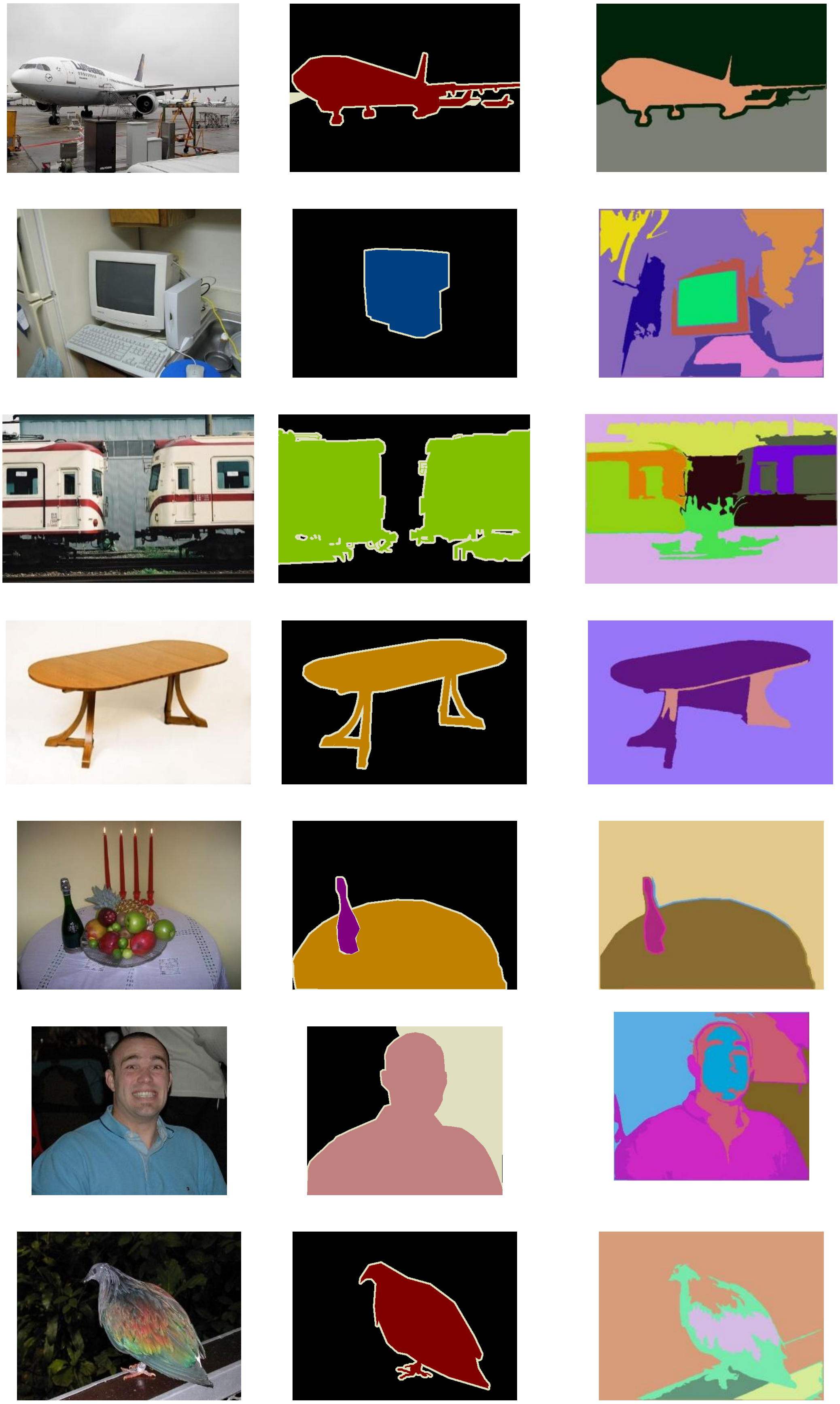


Figure 4.8: Result of Segmentation using the Proposed Technique on Pascal Dataset, Left: Original Image, Middle: Ground Truth, Right: Output

4.5.1 Time Comparison

Table 4.1 shows a comparison, in time, between the proposed technique and the original technique over sample images with different resolutions. As noted in Table 4.1, the time consumption for the proposed technique is much better than the original technique, especially with large images. In comparison between using the proposed technique to produce boundary-image or region-image, there is no clear differences in the time consumption. The rest of the experiments will be conducted for region-based, as thre is no differences in complexity and pascal's dataset provides ground truth for regions only. Table 4.2 shows a comparison, in time, between the proposed technique and the original technique over sample images from pascal. Overall, the proposed technique reduces running time by 58-66%.

Table 4.1: Time Comparison for the 1st Dataset

Image	Number of Pixels	K	Min. size	Time of Original Technique	Time of Proposed/ Boundaries	Time of Proposed/ Regions
Swan.jpg	154401	550	20	8 seconds	5 seconds	6 seconds
Duck.jpg	154401	550	25	8 seconds	5 seconds	5 seconds
India.jpg	154401	700	50	8 seconds	5 seconds	5 seconds
Bear.png	122694	1000	80	7 seconds	4 seconds	4 seconds
grain.gif	60000	1500	300	5 seconds	3 seconds	3 seconds

Table 4.2: Sample Time Comparison for Pascal Dataset

Image Name	Total Pixels	Time of Original Technique	Time of Proposed/ Regions	Running Time Difference (%)
629	115000	6.0	3.5	58.33
2007_000032	140500	8.0	5.0	62.50
918	160500	9.0	6.0	66.67
572	165500	9.0	6.0	66.67
241	166000	9.0	6.0	66.67
392	166000	9.0	6.0	66.67
738	166000	9.0	6.0	66.67
2007_000061	166500	9.0	6.0	66.67
243	166500	9.0	6.0	66.67
333	166500	9.0	6.0	66.67

4.5.2 Accuracy Comparison

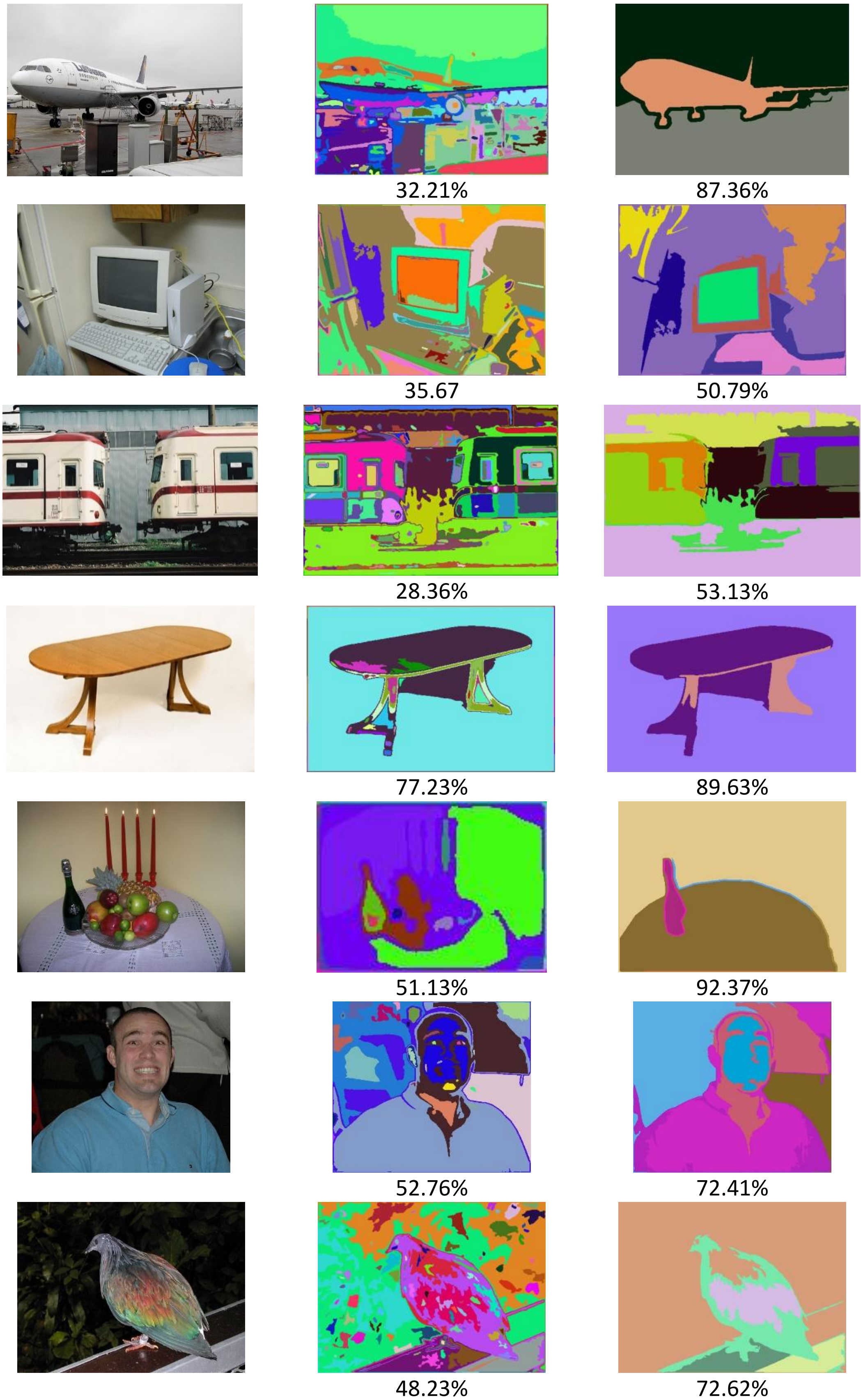
The accuracy of the segmentation techniques, is calculated by comparing the output segmented image with the ground truth, provided by the dataset, as calculated in Equation 4.1.

$$Accuracy = CorrectlySegmentedPixels/NumberPixels * 100\% \quad (4.1)$$

Table 4.3 gives a comparison between the accuracy of the proposed and compared technique based on a sub-set of pascal images. A wider comparison is given in Appendix A, Appendix B and Appendix C. Overall, the average accuracy result of testing 2000 images in pascal dataset is equal to 84.13627% ≈ 84.14% compared to 68.12% in the original technique. Examples of the results with accuracy percentage are given in Figure 4.9.

Table 4.3: Sample Accuracy Comparison for Pascal Dataset

Image Name	Total Pixels	# Correct in Original Approach	Accuracy of Original Approach	# Correct of Proposed Work	Accuracy of Proposed Work
629	115000	65421	56.888	86827	75.502
2007_000032	140500	75632	53.831	91827	65.357
918	160500	83254	51.872	123120	76.710
572	165500	116573	70.437	163448	98.760
241	166000	99932	60.200	147278	88.722
392	166000	59326	35.739	88668	53.414
738	166000	132658	79.914	157148	94.667
2007_000061	166500	103562	62.199	138246	83.031
243	166500	83253	50.002	111491	66.962
333	166500	82899	49.789	115226	69.205
346	166500	115382	69.298	151907	91.235
363	166500	147932	88.848	151077	90.737



Average	**46.51%**	**74.04%**

Figure 4.9: Result of Segmentation using Proposed and Original Techniques on Pascal Dataset, Left: Original Image, Middle: Original, Right: Proposed

4.6 Summary and Discussion

In this chapter, the validation and evaluation of the proposed graph-based image segmentation with initial points technique have been conducted, using a dataset of number of images with its ground truth, which describes, by any means, the correct segmentation of the image. The proposed algorithm is implemented and the output is compared to the ground truth. The accuracy measure, based on the differences between the produce output and the ground truth is calculated as an output for the evaluation process. The experimental results of the proposed technique have shown different statistics for the time consumption and accuracy. The proposed technique reduces running time by 58-66% and the average accuracy result of testing 2000 images in pascal dataset for the proposed technique is equal to 84.13627% ≈ 84.14% compared to 68.12% in the original technique. Overall, the proposed technique consumes less time and enhances the accuracy of the output.

Chapter Five
Conclusion and Future Work

In this chapter, conclusion and wrap up of this thesis are given. This chapter is organized as follows: Section 5.1 gives the conclusion of this thesis. Section 5.2 presents the limitation of the conducted research. Finally, future works are highlighted in Section 5.3.

5.1Conclusion

In this thesis, a graph-based image segmentation with initial points technique is presented. The selection of initial points is done depending on minimum edges weight, where the neighborhood pixels with low edge weight are grouped together to initialize a point, and then actual segments are created graph-based technique. The contributions of the proposed work are as follow:

- Modifying the process of graph-based segmentation technique and reduce the execution time by decreasing the number of examined.

- Perform graph-based image segmentation process around initial points with specific homogenous characteristics.

- Automatically define significant initial points in the image by examining the relationships between the neighborhood pixels and group similar pixels to form a point.

- Implement, evaluate and compare the results of the proposed work using an pascal image dataset ad proof the efficiency and accuracy of the proposed technique.

Overall, the proposed technique produced very accurate results compared to segmentation results generated using the original graph-based image segmentation on Pascal's Challenge dataset. The experimental results of the proposed technique have shown that the proposed technique reduces running time by 58-66% and the average accuracy result of testing 2000 images in pascal dataset for the proposed technique is equal to 84.13627% ≈ 84.14% compared to 68.12% in the original technique.

5.2 Limitations and Obstacles

The graph-based image segmentation depends on two parameters in order to be executed. These parameters cannot be predefined for all images and must be changed for each different image. The variable K, is a constant represents threshold used for the edge weight, where any edge weight value higher than K represent a boundary. The variable minSize, is a constant that represents the minimum number of pixels to be included in any segment. Experimentally, K and minSize were identified manually according to the number of objects in the image.

As the conducted experiments are designed for natural image, which include variety of colors, variations, texture and regions, the parameter values cannot be determined in advanced. Color variety in any segmentation technique leads to produce high number of regions. To minimize the number of regions, minimum acceptable number of pixels in single region, which is represented by the variable minSize, is increased. However, when the value of minSize is increased, some objects or sub-objects in the image may be lost since region of this item cannot reach the value of minSize threshold. Subsequently, setting up the parameters values is obstacle.

5.3 Future Work

There will be huge scope for further enhancement over the proposed technique in the future, some of which are listed below:

1- Find an automatic technique to assign values for the parameters K and minSize without any user interfere.

2- Instead of using a random color for each segment, create a selection technique to select a color that match the initial point color. Changing the colors of the segments to be from the original image will make the segments more mature allowing the viewer to perceive them correctly with naked eye.

3- Implements different smoothing level and allows for automatic selection of the the smoothing filters according to the resolution and colors level in each image.

4- Implement the technique on videos and medical images.

References

Bellis, Delphine. "The Perception of Spatial Depth in Kepler's and Descartes' Optics: A Study of an Epistemological Reversal." Boundaries, Extents and Circulations. Springer International Publishing, 2016. 125-152.

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). FCM: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, *10*(2-3), 191-203.

Bradley, P. S., & Fayyad, U. M. (1998, July). Refining Initial Points for K-Means Clustering. In *ICML* (Vol. 98, pp. 91-99).

Boykov, Y. Y., & Jolly, M. P. (2001). Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on* (Vol. 1, pp. 105-112). IEEE.

Burdescu, D. D., Ebânca, D. C., & Slabu, F. (2015). Graph-Based Segmentation Methods for Planar and Spatial Images. *IJCSA*, *12*(2), 120-143.

Chalasani, S. (2007). Graph Based Image Segmentation. *Tutorial-November-21*.

Corpet, F. (1988). Multiple sequence alignment with hierarchical clustering. *Nucleic acids research*, *16*(22), 10881-10890.

Da Silva, R. D., Schwartz, W. R., & Pedrini, H. (2011). Image segmentation based on wavelet feature descriptor and dimensionality reduction applied to remote sensing. *Chilean Journal of Statistics (ChJS)*, *2*(2).

Dagar, N. S., & Dahiya, P. K. (2016). Soft computing techniques for edge detection problem: a state-of-the-art review. *International Journal of Computer Applications*, *136*(12).

Dembele, D., & Kastner, P. (2003). Fuzzy C-means method for clustering microarray data. *Bioinformatics*, *19*(8), 973-980.

Dezso, B., R. Giachetta, I. Laszlo and I. Fekete, (2012). "Experimental study on graph-based image segmentation methods in the classification of satellite images", EARSeL eProceedings (ISSN 1729-3782), 11(1), 12–24.

Dhanachandra, N., Manglem, K., & Chanu, Y. J. (2015). Image segmentation using K-means clustering algorithm and subtractive clustering algorithm. *Procedia Computer Science*, *54*, 764-771.

Efford, Nick, (2000). Digital Image Processing: A Practical Introduction Using Java (with CD-ROM), Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA.

Fahim, A. M., Salem, A. M., Torkey, F. A., Saake, G., & Ramadan, M. A. (2009). An efficient k-means with good initial starting points. *Georgian Electronic Scientific Journal: Computer Science and Telecommunications*, *2*(19), 47-57.

Falcao, Alexandre, Spina, Thiago, Miranda, Paulo, Cappabianco, F´abio,(2010). Graph-Based Image Segmentation (Advanced tutorial), SIBGRAPI 2010, GRAMADO.RS.BRAZIL.

Fayyad, U. M., Reina, C., & Bradley, P. S. (1998, August). Initialization of Iterative Refinement Clustering Algorithms. In *KDD* (pp. 194-198).

Feinberg, Eugene, et al. "Content-aware digital media storage device and methods of using the same." U.S. Patent No. 9,448,918. 20 Sep. 2016.

Felzenszwalb, P. F., & Huttenlocher, D. P. (2004). Efficient graph-based image segmentation. *International journal of computer vision*, *59*(2), 167-181.

Ferrari, Stefano, (2011-2012). "Image segmentation – Image processing 1" Universit`a degli Studi di Milano

Garg, Ankit, (2016) "A Review on Image Segmentation Techniques" International Journal of Recent Research Aspects ISSN: 2349-7688, Special Issue: Conscientious and Unimpeachable Technologies 2016, pp. 53-55

Goel, L., Jain, N., & Srivastava, S. (2017, February). A novel PSO based algorithm to find initial seeds for the k-means clustering algorithm. In *Communication and Computing Systems: Proceedings of the International Conference on Communication and Computing Systems (ICCCS 2016), Gurgaon, India, 9-11 September, 2016* (p. 159). CRC Press.

Grundmann, M., Kwatra, V., Han, M., & Essa, I. (2010, June). Efficient hierarchical graph-based video segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on* (pp. 2141-2148). IEEE.

Guha, S., & Mishra, N. (2016). Clustering data streams. In *Data Stream Management* (pp. 169-187). Springer Berlin Heidelberg.

Hartigan, J. A., & Wong, M. A. (1979). Algorithm AS 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, *28*(1), 100-108.

Hickson, S., Birchfield, S., Essa, I., & Christensen, H. (2014). Efficient hierarchical graph-based segmentation of RGBD videos. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 344-351).

Holz, D., & Behnke, S. (2014). Approximate triangulation and region growing for efficient segmentation and smoothing of range images. *Robotics and Autonomous Systems*, *62*(9), 1282-1293.

Huang, Q. H., Lee, S. Y., Liu, L. Z., Lu, M. H., Jin, L. W., & Li, A. H. (2012). A robust graph-based segmentation method for breast tumors in ultrasound images. *Ultrasonics*, *52*(2), 266-275.

Islam, S., & Ahmed, M. (2013). Implementation of image segmentation for natural images using clustering methods. *Int J Emerg Technol Adv Eng*, *3*(3), 175-80.

Issac, A., Parthasarthi, M., & Dutta, M. K. (2015, February). An adaptive threshold based algorithm for optic disc and cup segmentation in fundus images. In *Signal Processing and Integrated Networks (SPIN), 2015 2nd International Conference on* (pp. 143-147). IEEE.

Jhaa, Sonu Kumar, Bannerjeeb, Purnendu, Banika, Subhadeep, (2013) "Random Walks Based Image Segmentation using Color Space Graphs" 1stInternational Conference on Computational Intelligence: Modelling, Techniques and Applications (CIMTA- 2013), Procedia Technology 10 (2013) 271 – 278

Kamdi, Shilpa, Krishna, R.K., "Image Segmentation and Region Growing Algorithm " International Journal of Computer Technology and Electronics Engineering (IJCTEE) (ISSN 2249-6343) Volume 2, Issue 1, 2012

Kaufman, L., & Rousseeuw, P. (1987). *Clustering by means of medoids*. North-Holland.

Khan, S. S., & Ahmad, A. (2004). Cluster center initialization algorithm for K-means clustering. *Pattern recognition letters*, *25*(11), 1293-1302.

Khurana, Deepika, Bhatia, M.P.S, "Analysis of Initial Centers for k-Means Clustering Algorithm" International Journal of Computer Applications (0975 – 8887). Volume 71– No.5, May 2013

Kumar, M. P., Torr, P. H., & Zisserman, A. (2010). Objcut: Efficient segmentation using top-down and bottom-up cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *32*(3), 530-545.

Li, L., Yao, J., Tu, J., Lu, X., Li, K., & Liu, Y. (2015, August). Edge-based split-and-merge superpixel segmentation. In *Information and Automation, 2015 IEEE International Conference on* (pp. 970-975). IEEE.

Martin, D. R., Fowlkes, C. C., & Malik, J. (2004). Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE transactions on pattern analysis and machine intelligence*, *26*(5), 530-549.

Meila, M., & Heckerman, D. (2013). An experimental comparison of several clustering and initialization methods. *arXiv preprint arXiv:1301.7401*.

Mobahi, H., Rao, S. R., Yang, A. Y., Sastry, S. S., & Ma, Y. (2011). Segmentation of natural images by texture and boundary compression. *International journal of computer vision*, *95*(1), 86-98.

Mottaghi, R., Chen, X., Liu, X., Cho, N. G., Lee, S. W., Fidler, S., ... & Yuille, A. (2014). The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 891-898).

Nazeer, K. A., & Sebastian, M. P. (2009, July). Improving the Accuracy and Efficiency of the k-means Clustering Algorithm. *In Proceedings of the World congress on Engineering* (Vol. 1, pp. 1-3).

Narang, Sahil, Rathinavel, Kishore. "An implementation of Efficient Graph-Based Image Segmentation". University of North Carolina, Chapel Hill

Pawan, M., Kumar, D. Koller, (2010) "Region Selection for Scene Understanding". http://mpawankumar.info/research/region-selection.html

Ren, X., Fowlkes, C. C., & Malik, J. (2006, May). Figure/ground assignment in natural images. In *Proceedings of the 9th European conference on Computer Vision-Volume Part II* (pp. 614-627). Springer-Verlag.

Rudra, A. K., Chowdhury, A. S., Elnakib, A., Khalifa, F., Soliman, A., Beache, G., & El-Baz, A. (2013). Kidney segmentation using graph cuts and pixel connectivity. *Pattern Recognition Letters*, *34*(13), 1470-1475.

Rungruangbaiyok, S., Duangsoithong, R., & Chetpattananondh, K. (2015, June). Ensemble threshold segmentation for hand detection. In *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2015 12th International Conference on* (pp. 1-5). IEEE.

Saini, Sujata, Arora, Komal. (2014) "A Study Analysis on the Different Image Segmentation Techniques" International Journal of Information & Computation Technology. ISSN 0974-2239 Volume 4, Number 14 (2014), pp. 1445-1452

Sarfraz, Muhammad, (2014). Computer vision and image processing in intelligent systems and multimedia technologies, United states of America, Information Sciences Reference.

Semwal, Ashish, et al. "A SURVEY: On Image Segmentation and its Various Techniques.", International Research Journal of Engineering and Technology, 3 (12), 1565-1568 (2016).

Singh, Onkr, (2011). "New Method of Image Segmentation for Standard Images", IJCST Vol. 2, Issue 3, September 2011 I SSN: 2 2 2 9 - 4 3 3 3 (Print) | ISSN : 0 9 7 6 - 8 4 9 1 (Online)

Shafeeq, A., & Hareesha, K. S. (2012). Dynamic clustering of data with modified k-means algorithm. In *Proceedings of the 2012 conference on information and computer networks* (pp. 221-225).

Sonka, Milan, Hlavac, Vaclav, Boyle, Roger, (2014). Image Processing, analysis, and machine vision, Global Engineering, United States of America

Stephanakis, Ioannis M., Anastassopoulos, George C., (2006). "SEGMENTATION USING ADAPTIVE THRESHOLDING OF THE IMAGE HISTOGRAM ACCORDING TO THE INCREMENTAL RATES OF THE SEGMENT LIKELIHOOD FUNCTIONS" “unpublished”, pp. 1-5.

Subasic, P., Coloma, K., Zhang, G., Chang, J., & Shukla, M. (2014). *U.S. Patent No. 8,793,252*. Washington, DC: U.S. Patent and Trademark Office.

Tishby, N., & Slonim, N. (2001). Data clustering by markovian relaxation and the information bottleneck method. In *Advances in neural information processing systems* (pp. 640-646).

Umbaugh, Scott, (2010). Digital Image Processing and Analysis: Human and Computer Vision Application with CVIP tools, CRC Press, Tayloee & Francis Group.

Vala, M. H. J., & Baxi, A. (2013). A review on Otsu image segmentation algorithm. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, *2*(2), pp-387.

Visual Object Classes Challenge 2012 (VOC2012), http://host.robots.ox.ac.uk/pascal/VOC/voc2012/index.html

Vu, N. B. S. (2008). *Image segmentation with semantic priors: A graph cut approach*. University of California, Santa Barbara.

Yang, J., Gan, Z., Li, K., & Hou, C. (2015). Graph-based segmentation for RGB-D data using 3-D geometry enhanced superpixels. *IEEE transactions on cybernetics*, *45*(5), 927-940.

Wassenberg, J., Middelmann, W., & Sanders, P. (2009, September). An efficient parallel algorithm for graph-based image segmentation. In *International Conference on Computer Analysis of Images and Patterns* (pp. 1003-1010). Springer, Berlin, Heidelberg.

Wojnar, L. (1998). *Image analysis: applications in materials engineering*. Crc Press.

Yogamangalam, R., & Karthikeyan, B. (2013). Segmentation techniques comparison in image processing. *International Journal of Engineering and Technology (IJET)*, *5*(1), 307-313.

Zhang, Lixia, et al. "Adaptive image segmentation based on color clustering for person re-identification." Soft Computing (2016): 1-11.

Zhang, F., & Hancock, E. R. (2008). Graph spectral image smoothing using the heat kernel. *Pattern Recognition*, *41*(11), 3328-3342.

Zheng, Y., Jeon, B., Xu, D., Wu, Q. M., & Zhang, H. (2015). Image segmentation by generalized hierarchical fuzzy C-means algorithm. *Journal of Intelligent & Fuzzy Systems*, *28*(2), 961-973.

Appendix A: Original Segmentation Results

Image Name	W * H		Total Pixels	Correctly Segmented Pixels	Wrong Segmented Pixels	Accuracy %
032	500	281	140500	75632	64868	**53.83060498**
033	500	366	183000	87326	95674	**47.71912568**
039	500	375	187500	135848	51652	**72.45226667**
042	500	335	167500	159638	7862	**95.30626866**
061	500	333	166500	103562	62938	**62.1993994**
063	500	375	187500	136582	50918	**72.84373333**
068	500	375	187500	180031	7469	**96.01653333**
121	500	375	187500	134862	52638	**71.9264**
123	500	375	187500	129732	57768	**69.1904**
129	500	375	187500	167432	20068	**89.29706667**
170	500	375	187500	162148	25352	**86.47893333**
175	500	375	187500	135925	51575	**72.49333333**
187	500	375	187500	98237	89263	**52.39306667**
241	500	332	166000	99932	66068	**60.2**
243	500	333	166500	83253	83247	**50.0018018**
250	500	375	187500	132481	55019	**70.65653333**
256	500	343	171500	135982	35518	**79.28979592**
323	500	375	187500	87230	100270	**46.52266667**
332	500	375	187500	170316	17184	**90.8352**
333	500	333	166500	82899	83601	**49.78918919**
346	500	333	166500	115382	51118	**69.2984985**
363	500	333	166500	147932	18568	**88.84804805**
364	500	375	187500	123862	63638	**66.05973333**
392	500	332	166000	59326	106674	**35.73855422**
452	500	375	187500	143298	44202	**76.4256**
464	500	375	187500	143293	44207	**76.42293333**
480	500	375	187500	100682	86818	**53.69706667**
491	500	334	167000	153249	13751	**91.76586826**
504	500	412	206000	162329	43671	**78.80048544**
515	500	375	187500	136297	51203	**72.69173333**
528	500	375	187500	153268	34232	**81.74293333**
529	500	375	187500	163288	24212	**87.08693333**
549	500	375	187500	127320	60180	**67.904**
559	500	370	185000	107068	77932	**57.87459459**
572	500	331	165500	116573	48927	**70.43685801**
584	500	375	187500	139329	48171	**74.3088**
629	500	230	115000	65421	49579	**56.88782609**
636	500	335	167500	143219	24281	**85.5038806**
645	500	375	187500	143872	43628	**76.73173333**
648	500	333	166500	147953	18547	**88.86066066**
661	500	375	187500	153279	34221	**81.7488**

Image Name	W * H		Total Pixels	Correctly Segmented Pixels	Wrong Segmented Pixels	Accuracy %
663	500	422	211000	183273	27727	**86.85924171**
676	500	375	187500	142732	44768	**76.12373333**
713	500	336	168000	160007	7993	**95.2422619**
720	500	333	166500	128185	38315	**76.98798799**
727	500	375	187500	113938	73562	**60.76693333**
733	450	435	195750	103279	92471	**52.76066411**
738	500	332	166000	132658	33342	**79.91445783**
762	500	375	187500	147003	40497	**78.4016**
768	500	375	187500	106832	80668	**56.97706667**
793	500	375	187500	123589	63911	**65.91413333**
821	500	375	187500	142638	44862	**76.0736**
822	500	375	187500	134421	53079	**71.6912**
825	500	375	187500	152486	35014	**81.32586667**
832	500	375	187500	143296	44204	**76.42453333**
839	500	370	185000	110385	74615	**59.66756757**
842	500	375	187500	132658	54842	**70.75093333**
843	500	375	187500	143288	44212	**76.42026667**
846	500	375	187500	153262	34238	**81.73973333**
847	500	333	166500	79898	86602	**47.98678679**
848	500	375	187500	123687	63813	**65.9664**
855	500	375	187500	110384	77116	**58.87146667**
856	500	375	187500	103289	84211	**55.08746667**
859	500	375	187500	99973	87527	**53.31893333**
863	500	375	187500	113279	74221	**60.41546667**
864	500	421	210500	135769	74731	**64.49833729**
868	500	375	187500	168273	19227	**89.7456**
870	500	375	187500	167923	19577	**89.55893333**
871	500	375	187500	161109	26391	**85.9248**
876	500	375	187500	163289	24211	**87.08746667**
877	500	450	225000	135892	89108	**60.39644444**
878	500	375	187500	113298	74202	**60.4256**
892	500	375	187500	123595	63905	**65.91733333**
896	500	375	187500	139842	47658	**74.5824**
903	500	440	220000	87368	132632	**39.71272727**
905	500	375	187500	103983	83517	**55.4576**
906	500	375	187500	123792	63708	**66.0224**
908	500	375	187500	143280	44220	**76.416**
911	500	350	175000	93182	81818	**53.24685714**
912	500	375	187500	113789	73711	**60.68746667**
915	500	375	187500	98322	89178	**52.4384**
917	500	375	187500	139872	47628	**74.5984**
918	500	321	160500	83254	77246	**51.87165109**
920	500	375	187500	103267	84233	**55.07573333**

Image Name	W * H		Total Pixels	Correctly Segmented Pixels	Wrong Segmented Pixels	Accuracy %
923	500	375	187500	99732	87768	**53.1904**
925	500	375	187500	158732	28768	**84.65706667**
927	500	375	187500	127953	59547	**68.2416**
930	500	333	166500	79832	86668	**47.94714715**
931	500	375	187500	147213	40287	**78.5136**
934	500	375	187500	128582	58918	**68.57706667**
935	500	375	187500	56598	130902	**30.1856**
937	500	375	187500	83265	104235	**44.408**
942	500	421	210500	99731	110769	**47.37814727**
943	500	375	187500	132018	55482	**70.4096**
948	500	375	187500	113294	74206	**60.42346667**
963	500	350	175000	109723	65277	**62.69885714**
967	500	375	187500	147932	39568	**78.89706667**
968	500	375	187500	103548	83952	**55.2256**
975	500	375	187500	123498	64002	**65.8656**
					Average	**68.42676**

Appendix B: Proposed Work Results

Image Name	W * H		Total Pixels	Correctly Segmented Pixels	Wrongly Segmented Pixels	Accuracy %
032	500	281	140500	91827	48673	65.35729537
033	500	366	183000	111307	71693	60.82349727
039	500	375	187500	153448	34052	81.83893333
042	500	335	167500	162114	5386	96.78447761
061	500	333	166500	138246	28254	83.03063063
063	500	375	187500	179932	7568	95.96373333
068	500	375	187500	186624	876	99.5328
121	500	375	187500	178621	8879	95.26453333
123	500	375	187500	161122	26378	85.93173333
129	500	375	187500	178640	8860	95.27466667
170	500	375	187500	168932	18568	90.09706667
175	500	375	187500	170003	17497	90.66826667
187	500	375	187500	127234	60266	67.85813333
241	500	332	166000	147278	18722	88.72168675
243	500	333	166500	111491	55009	66.96156156
250	500	375	187500	162130	25370	86.46933333
256	500	343	171500	153681	17819	89.60991254
323	500	375	187500	101982	85518	54.3904
332	500	375	187500	186618	882	99.5296
333	500	333	166500	115226	51274	69.2048048
346	500	333	166500	151907	14593	91.23543544
363	500	333	166500	151077	15423	90.73693694
364	500	375	187500	142598	44902	76.05226667
392	500	332	166000	88668	77332	53.41445783
452	500	375	187500	172174	15326	91.82613333
464	500	375	187500	167659	19841	89.41813333
480	500	375	187500	128550	58950	68.56
491	500	334	167000	166141	859	99.48562874
504	500	412	206000	192116	13884	93.26019417
515	500	375	187500	173303	14197	92.42826667
528	500	375	187500	179929	7571	95.96213333
529	500	375	187500	180044	7456	96.02346667
549	500	375	187500	182646	4854	97.4112
559	500	370	185000	169755	15245	91.75945946
572	500	331	165500	163448	2052	98.76012085
584	500	375	187500	181918	5582	97.02293333
629	500	230	115000	86827	28173	75.50173913
636	500	335	167500	156615	10885	93.50149254
645	500	375	187500	157428	30072	83.9616
648	500	333	166500	161211	5289	96.82342342
661	500	375	187500	172888	14612	92.20693333

Image Name	W * H		Total Pixels	Correctly Segmented Pixels	Wrongly Segmented Pixels	Accuracy %
663	500	422	211000	209743	1257	99.4042654
676	500	375	187500	184318	3182	98.30293333
713	500	336	168000	167131	869	99.4827381
720	500	333	166500	130689	35811	78.49189189
727	500	375	187500	124534	62966	66.41813333
733	450	435	195750	139806	55944	71.42068966
738	500	332	166000	157148	8852	94.66746988
762	500	375	187500	169592	17908	90.44906667
768	500	375	187500	180101	7399	96.05386667
793	500	375	187500	176253	11247	94.0016
821	500	375	187500	163591	23909	87.24853333
822	500	375	187500	179532	7968	95.7504
825	500	375	187500	170063	17437	90.70026667
832	500	375	187500	162060	25440	86.432
839	500	370	185000	143259	41741	77.4372973
842	500	375	187500	169755	17745	90.536
843	500	375	187500	163448	24052	87.17226667
846	500	375	187500	181918	5582	97.02293333
847	500	333	166500	86827	79673	52.14834835
848	500	375	187500	167131	20369	89.13653333
855	500	375	187500	130689	56811	69.7008
856	500	375	187500	124534	62966	66.41813333
859	500	375	187500	139806	47694	74.5632
863	500	375	187500	138246	49254	73.7312
864	500	421	210500	179932	30568	85.4783848
868	500	375	187500	176624	10876	94.19946667
870	500	375	187500	178621	8879	95.26453333
871	500	375	187500	161122	26378	85.93173333
876	500	375	187500	178640	8860	95.27466667
877	500	450	225000	183262	41738	81.44977778
878	500	375	187500	162358	25142	86.59093333
892	500	375	187500	159731	27769	85.18986667
896	500	375	187500	153269	34231	81.74346667
903	500	440	220000	173268	46732	78.75818182
905	500	375	187500	162437	25063	86.63306667
906	500	375	187500	172001	15499	91.73386667
908	500	375	187500	173265	14235	92.408
911	500	350	175000	134892	40108	77.08114286
912	500	375	187500	153291	34209	81.7552
915	500	375	187500	136581	50919	72.8432
917	500	375	187500	162579	24921	86.7088
918	500	321	160500	123120	37380	76.71028037
920	500	375	187500	157892	29608	84.20906667

Image Name	W * H		Total Pixels	Correctly Segmented Pixels	Wrongly Segmented Pixels	Accuracy %
923	500	375	187500	143692	43808	76.63573333
925	500	375	187500	180034	7466	96.01813333
927	500	375	187500	173292	14208	92.4224
930	500	333	166500	113256	53244	68.02162162
931	500	375	187500	168923	18577	90.09226667
934	500	375	187500	175315	12185	93.50133333
935	500	375	187500	91827	95673	48.9744
937	500	375	187500	111307	76193	59.36373333
942	500	421	210500	153448	57052	72.89691211
943	500	375	187500	162114	25386	86.4608
948	500	375	187500	138246	49254	73.7312
963	500	350	175000	132592	42408	75.76685714
967	500	375	187500	186624	876	99.5328
968	500	375	187500	163289	24211	87.08746667
975	500	375	187500	142418	45082	75.95626667
Average						84.56344593

Appendix C: Comparison Results

Image Name	H	W	Total Pixels	# Correct in Original Approach	Accuracy Percentage of Original Approach	# Correct of Proposed Work	Accuracy Percentage of Proposed Work	Running Time using Original Approach	Running Time using Proposed Work	Running Time Difference (%)
629	500	230	115000	65421	56.888	86827	75.502	6.0	3.5	58.33
2007_000032	500	281	140500	75632	53.831	91827	65.357	8.0	5.0	62.50
918	500	321	160500	83254	51.872	123120	76.710	9.0	6.0	66.67
572	500	331	165500	116573	70.437	163448	98.760	9.0	6.0	66.67
241	500	332	166000	99932	60.200	147278	88.722	9.0	6.0	66.67
392	500	332	166000	59326	35.739	88668	53.414	9.0	6.0	66.67
738	500	332	166000	132658	79.914	157148	94.667	9.0	6.0	66.67
2007_000061	500	333	166500	103562	62.199	138246	83.031	9.0	6.0	66.67
243	500	333	166500	83253	50.002	111491	66.962	9.0	6.0	66.67
333	500	333	166500	82899	49.789	115226	69.205	9.0	6.0	66.67
346	500	333	166500	115382	69.298	151907	91.235	9.0	6.0	66.67
363	500	333	166500	147932	88.848	151077	90.737	9.0	6.0	66.67
648	500	333	166500	147953	88.861	161211	96.823	9.0	6.0	66.67
720	500	333	166500	128185	76.988	130689	78.492	9.0	6.0	66.67
847	500	333	166500	79898	47.987	86827	52.148	9.0	6.0	66.67
930	500	333	166500	79832	47.947	113256	68.022	9.0	6.0	66.67
491	500	334	167000	153249	91.766	166141	99.486	10.0	6.5	65.00
2007_000042	500	335	167500	159638	95.306	162114	96.784	10.0	6.5	65.00
636	500	335	167500	143219	85.504	156615	93.501	10.0	6.5	65.00
713	500	336	168000	160007	95.242	167131	99.483	10.0	6.5	65.00
256	500	343	171500	135982	79.290	153681	89.610	11.0	7.0	63.64
911	500	350	175000	93182	53.247	134892	77.081	11.5	7.0	60.87
963	500	350	175000	109723	62.699	132592	75.767	11.5	7.0	60.87
2007_000033	500	366	183000	87326	47.719	111307	60.823	12.0	7.5	62.50
559	500	370	185000	107068	57.875	169755	91.759	12.0	7.5	62.50
839	500	370	185000	110385	59.668	143259	77.437	12.0	7.5	62.50
2007_000039	500	375	187500	135848	72.452	153448	81.839	13.0	8.0	61.54
2007_000063	500	375	187500	136582	72.844	179932	95.964	13.0	8.0	61.54
2007_000068	500	375	187500	180031	96.017	186624	99.533	13.0	8.0	61.54
2007_000121	500	375	187500	134862	71.926	178621	95.265	13.0	8.0	61.54
123	500	375	187500	129732	69.190	161122	85.932	13.0	8.0	61.54
129	500	375	187500	167432	89.297	178640	95.275	13.0	8.0	61.54
170	500	375	187500	162148	86.479	168932	90.097	13.0	8.0	61.54

Image Name	H	W	Total Pixels	# Correct in Original Approach	Accuracy Percentage of Original Approach	# Correct of Proposed Work	Accuracy Percentage of Proposed Work	Running Time using Original Approach	Running Time using Proposed Work	Running Time Difference (%)
175	500	375	187500	135925	72.493	170003	90.668	13.0	8.0	61.54
187	500	375	187500	98237	52.393	127234	67.858	13.0	8.0	61.54
250	500	375	187500	132481	70.657	162130	86.469	13.0	8.0	61.54
323	500	375	187500	87230	46.523	101982	54.390	13.0	8.0	61.54
332	500	375	187500	170316	90.835	186618	99.530	13.0	8.0	61.54
364	500	375	187500	123862	66.060	142598	76.052	13.0	8.0	61.54
452	500	375	187500	143298	76.426	172174	91.826	13.0	8.0	61.54
464	500	375	187500	143293	76.423	167659	89.418	13.0	8.0	61.54
480	500	375	187500	100682	53.697	128550	68.560	13.0	8.0	61.54
515	500	375	187500	136297	72.692	173303	92.428	13.0	8.0	61.54
528	500	375	187500	153268	81.743	179929	95.962	13.0	8.0	61.54
529	500	375	187500	163288	87.087	180044	96.023	13.0	8.0	61.54
549	500	375	187500	127320	67.904	182646	97.411	13.0	8.0	61.54
584	500	375	187500	139329	74.309	181918	97.023	13.0	8.0	61.54
645	500	375	187500	143872	76.732	157428	83.962	13.0	8.0	61.54
661	500	375	187500	153279	81.749	172888	92.207	13.0	8.0	61.54
676	500	375	187500	142732	76.124	184318	98.303	13.0	8.0	61.54
727	500	375	187500	113938	60.767	124534	66.418	13.0	8.0	61.54
762	500	375	187500	147003	78.402	169592	90.449	13.0	8.0	61.54
768	500	375	187500	106832	56.977	180101	96.054	13.0	8.0	61.54
793	500	375	187500	123589	65.914	176253	94.002	13.0	8.0	61.54
821	500	375	187500	142638	76.074	163591	87.249	13.0	8.0	61.54
822	500	375	187500	134421	71.691	179532	95.750	13.0	8.0	61.54
825	500	375	187500	152486	81.326	170063	90.700	13.0	8.0	61.54
832	500	375	187500	143296	76.425	162060	86.432	13.0	8.0	61.54
842	500	375	187500	132658	70.751	169755	90.536	13.0	8.0	61.54
843	500	375	187500	143288	76.420	163448	87.172	13.0	8.0	61.54
846	500	375	187500	153262	81.740	181918	97.023	13.0	8.0	61.54
848	500	375	187500	123687	65.966	167131	89.137	13.0	8.0	61.54
855	500	375	187500	110384	58.871	130689	69.701	13.0	8.0	61.54
856	500	375	187500	103289	55.087	124534	66.418	13.0	8.0	61.54
859	500	375	187500	99973	53.319	139806	74.563	13.0	8.0	61.54
863	500	375	187500	113279	60.415	138246	73.731	13.0	8.0	61.54
868	500	375	187500	168273	89.746	176624	94.199	13.0	8.0	61.54
870	500	375	187500	167923	89.559	178621	95.265	13.0	8.0	61.54

Image Name	H	W	Total Pixels	# Correct in Original Approach	Accuracy Percentage of Original Approach	# Correct of Proposed Work	Accuracy Percentage of Proposed Work	Running Time using Original Approach	Running Time using Proposed Work	Running Time Difference (%)
871	500	375	187500	161109	85.925	161122	85.932	13.0	8.0	61.54
876	500	375	187500	163289	87.087	178640	95.275	13.0	8.0	61.54
878	500	375	187500	113298	60.426	162358	86.591	13.0	8.0	61.54
892	500	375	187500	123595	65.917	159731	85.190	13.0	8.0	61.54
896	500	375	187500	139842	74.582	153269	81.743	13.0	8.0	61.54
905	500	375	187500	103983	55.458	162437	86.633	13.0	8.0	61.54
906	500	375	187500	123792	66.022	172001	91.734	13.0	8.0	61.54
908	500	375	187500	143280	76.416	173265	92.408	13.0	8.0	61.54
912	500	375	187500	113789	60.687	153291	81.755	13.0	8.0	61.54
915	500	375	187500	98322	52.438	136581	72.843	13.0	8.0	61.54
917	500	375	187500	139872	74.598	162579	86.709	13.0	8.0	61.54
920	500	375	187500	103267	55.076	157892	84.209	13.0	8.0	61.54
923	500	375	187500	99732	53.190	143692	76.636	13.0	8.0	61.54
925	500	375	187500	158732	84.657	180034	96.018	13.0	8.0	61.54
927	500	375	187500	127953	68.242	173292	92.422	13.0	8.0	61.54
931	500	375	187500	147213	78.514	168923	90.092	13.0	8.0	61.54
934	500	375	187500	128582	68.577	175315	93.501	13.0	8.0	61.54
935	500	375	187500	56598	30.186	91827	48.974	13.0	8.0	61.54
937	500	375	187500	83265	44.408	111307	59.364	13.0	8.0	61.54
943	500	375	187500	132018	70.410	162114	86.461	13.0	8.0	61.54
948	500	375	187500	113294	60.423	138246	73.731	13.0	8.0	61.54
967	500	375	187500	147932	78.897	186624	99.533	13.0	8.0	61.54
968	500	375	187500	103548	55.226	163289	87.087	13.0	8.0	61.54
975	500	375	187500	123498	65.866	142418	75.956	13.0	8.0	61.54
733	450	435	195750	103279	52.761	139806	71.421	14.0	9.0	64.29
504	500	412	206000	162329	78.800	192116	93.260	14.0	9.0	64.29
864	500	421	210500	135769	64.498	179932	85.478	15.0	9.5	63.33
942	500	421	210500	99731	47.378	153448	72.897	15.0	9.5	63.33
663	500	422	211000	183273	86.859	209743	99.404	15.0	9.5	63.33
903	500	440	220000	87368	39.713	173268	78.758	16.0	10.0	62.50
877	500	450	225000	135892	60.396	183262	81.450	16.0	10.0	62.50
					68.427		**84.563**			**62.55**