Rochester Institute of Technology

RIT Scholar Works

8-2007

Automatic image segmentation by dynamic region growth and multiresolution merging

Luis Enrique Garcia Ugarriza

Automatic Image Segmentation by Dynamic Region Growth and Multiresolution Merging

by

Luis Enrique Garcia Ugarriza

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of
Master of Science
in
Electrical Engineering

Approved By:

Eli Saber

Dr. Eli Saber
Electrical Engineering
Primary Adviser

Vincent J. Amuso

Dr. Vincent Amuso
Electrical Engineering
Department Head

Sohail Dianat

Dr. Sohail Dianat
Electrical Engineering

Imaging Science Laboratory
Department of Electrical Engineering
Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, New York
August 2007

Thesis Release Permission Form

Rochester Institute of Technology

Kate Gleason College of Engineering

Title: Automatic Image Segmentation by Dynamic Region Growth and Multiresolution Merging

I, Luis Enrique Garcia Ugarriza, hereby grant permission to the Wallace Memorial Library reporduce my thesis in whole or part.

Luis Enrique Garcia Ugarriza
_____
Luis Enrique Garcia Ugarriza

08/03/2007
_____
Date

Dedication

This thesis is dedicated to my family.

To my wife, Dr. Mary Mulcahey, for her continued support, patience, and inspiration.

To my father, Luis Garcia, who passed on to me his view that everything is possible.

To my mother, Lucy Ugarriza, for her never-failing love.

To my sisters, Monica and Tatiana, for always being there for me.

Acknowledgments

I would like to thank Professor Eli Saber, who gave me the opportunity to work on this exciting project. Dr. Saber instructed me in the art of image processing, and also taught me valuable lessons on life, such as responsibility and integrity. He showed me the importance of trying all possible approaches before asking questions. He showed me that there are no bad ideas, just incorrect ways of promoting them. Most importantly, he inspired me to pursue perfection and to always search for the next frontier. I also thank my co-advisor, Dr. Vincent Amuso, without whose guidance, reviews, and recommendations this project would not have come to its level of quality and importance. He has always been there to talk about my ideas, to proofread and mark up my papers and chapters, and to ask me good questions to help me think through my problems, whether analytical, computational, or even religious.

Besides my advisors, I thank the rest of the people that I have approached for different questions: Professor Sohail Dianat, who provided insightful knowledge for any question I would unexpectedly ask him; Professor Eric Peskin, for his continued insistency on improving the quality of segmentation; Mark Shaw, for his support and trust in the work provided; and to all my colleagues, Mustafa Jaber, Sreenivas Patil, Manoj Reddy, Narne Harsha and Guru Balasubramanian, who made the "HP lab" a fun place to be during the many days and nights we spent together.

A special thanks goes to my friend, Professor Daniel Phillips, who is most responsible for my choosing the field of signal processing. He showed me the fun of searching for answers, the thrill of discovering new methods to achieve a purpose, and the satisfaction of developing a useful and relevant invention for tomorrow.

Abstract

Image segmentation is a fundamental task in many computer vision applications. We present a novel unsupervised color image segmentation algorithm named GSEG, which exploits the information obtained from detecting edges in color images. By using a color gradient detection technique, pixels without edges are clustered and labeled individually to identify the image content. Elements that contain higher gradient density are included by a dynamic generation of clusters as the segmentation progresses. By quantizing the colors in the image and extracting texture information from the neighborhood entropy of each pixel, the proposed method obtains accurate models of texture that are highly effective to merge regions that belong to the same object . Experimental results for various image scenarios in comparison with state-of-the-art segmentation techniques demonstrate the performance advantages of the proposed method.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In recent years, automatic image segmentation has become a prominent objective in image analysis and computer vision. Image segmentation is defined as the classification of all the picture elements or pixels in an image into different clusters that exhibit similar features. Various image segmentation techniques have been proposed in past literature, where color, edges, and texture are used as properties for classification. Using these properties, the images can be analyzed for use in several applications including video surveillance, image retrieval, medical imaging analysis, and object classification.

Initially, segmentation algorithms [1] were implemented using only gray levels, yielding regions without any significant information concerning the content of the image. The advancement in color technology helped in obtaining meaningful color segmentation of images as described in [2, 3]. Color provided definite advantages over gray-level segmentations, but these procedures consisted of only clustering pixels by the similarity of colors. Location of pixels was not taken into account and, therefore, regions did not display the compactness of objects containing varying colors throughout. This realization was the beginning of a series of challenges that have proven to be some of the most difficult and important steps in achieving effective image understanding. Different approaches have been introduced to improve on the shortcomings of past algorithms.

The initial step is to assure that each region is not only clustered based on the color similarity of pixels but also on other characteristics (e.g. location and distribution). This

problem was confronted in the gray-level domain in [4] by implementing the k-means clustering algorithm and Gibbs random-field model to obtain spatially contiguous regions. This method was extended for multichannel images by Chang, et al, [5] by assuming that each individual channel is conditionally independent. A good segmentation technique requires that each region is bounded by continuous edges that separate individual objects in the image. Saber, et al, dealt with this problem by extending the algorithm in [5] with a vector-edge field and a split-and-merge procedure to obtain an improved segmentation and linked-edge map. Images can have different content and cannot be restricted to a fixed number of regions. The work in [6] uses a predetermined number of clusters at the beginning to yield the final segmentation map. To this effect, the algorithm forces any type of scenario to fit into a set number of clusters, yielding an incoherent segmentation map in some cases. Therefore, it is necessary to develop a technique to choose the number of clusters for each image prior to segmentation.

Jianping et al. [7] proposed a method to select the number of clusters in an image by acquiring the location of clusters between adjacent-edge regions. The challenge in this method is to evaluate the correct threshold that differentiates edge pixels with false-edge pixels. Jianping attempts to solve this problem using a fast, entropic-threshold technique in a second-order neighborhood of each pixel. However, the final segmentation map obtained by this procedure does not yield meaningful continuous edges. A different approach to define the number of clusters needed was instituted by Wan et al. [8] who developed a set of rules that split or merge the clusters to obtain a final segmentation map with individual meaningful regions.

By combining initial clustered regions, Liu et al. [9] proposed an approach to find the best match with the predetermined shapes to obtain a final segmentation. There are two problems with this approach: first, the algorithm always assumes that the initial segmentation has a larger number of clusters than required and proceeds with this assumption to yield the final segmentation map. However, this is not true in every case, rendering this assumption inconsistent. Second, the combination of over-segmented regions makes the

2

algorithm vulnerable to match with incorrect shapes. On the other hand, D'Elia et al. [10] initiates the proposed algorithm by considering the entire image as a single region and obtains the final segmentation map by combining the Bayesian classifier and the Split-and-Merge technique. This approach solves the problem of identifying the correct number of initial clusters by investigating individual regions for further segmentation. The major setback of the Bayesian approach is that it yields too many segments in a pattern or texture region, which, in turn, produces a cluttered final segmentation map.

The task of segmenting images containing texture is an ongoing area of research in image processing. Derin et al. [11] proposed a technique of comparing the Gibbs distribution results to known textures. This technique was not applicable in the presence of noise and proved to be computationally prohibiting. It became obvious that an alternate scheme was required to identify patterns in images. Pyramid-structured wavelet transforms first appeared in the work of Mallat et al.[12] and has become an important alternate approach to identify patterns. Unser et al. [13] uses a variation of the discrete wavelet transform for characterizing texture properties. In this study they limited their detection to a set of 12 Brodatz textures. Further analysis of wavelet theory was completed by Fan et al. [14], who showed that in natural textures the periodicity results in dependencies across the discrete wavelet transform subbands. This approach was extended to 55 Brodatz textures. Chen et al. [15] modified the approach of identifying textures without using Brodatz models by simply classifying texture into a limited number of classes: smooth, horizontal, vertical, and complex. This simplification made this approach difficult for images that contain multiple textures that share common boundaries. Deng et al. [16] extended the work with an automated method to identify texture regions without prior models of textures. This method first quantizes the image into a few colors and filters them individually to determine the smoothness of the local areas. The use of color quantization by this study causes a major problem when regions of varying shades, due to illumination, appear in the image. For instance, the sky in panoramic images may change from light blue to dark blue in a smooth transition, displaying no clear boundary of the sky. However, the quantization of

colors will often generate clusters for each shade of blue, which ultimately over-segments the image.

We propose a segmentation algorithm that automatically selects clusters for images and characterizes the texture present in each cluster to obtain the final segmentation. Using an edge-color detection algorithm [17], which provides varying edge values according to color differences, our procedure first created clusters on locations without edges. All the pixels in a cluster are given a label, and the collection of these labeled pixels is referred to as seeds. A limited area of an image is selected at this stage. The remaining area is segmented by both growth of existent seeds and generation of new seeds, which is performed at increasingly higher values of color changes until all the pixels have received a segmentation label. The characterization of texture is performed by quantizing the color in the image and evaluating the entropic color information present at each seed. The seeds that are part of textures have similar values of color entropy and consequently are merged to obtain the final segmentation.

This effective procedure takes into account the fact that segmentation is a low-level process and, as such, should not require a large amount of computational complexity. No training or prior knowledge of the input image is part of the algorithm. The algorithm is compiled in a MATLAB environment over a large database of diverse images. Compared to other segmentation algorithms, and using the Berkeley manually segmented database as a ground truth, the proposed algorithm consistently outperforms the segmentation results from other segmentation techniques.

In chapter 2, a review of the necessary background required to effectively implement and test our algorithm is presented. Our proposed algorithm is subdivided into three sections: 3.1 introduces the Region Growth and Dynamic Seed procedure, 3.2 explains our approach to characterize texture present on images, and 3.3 provides the methodology used in our novel multiresolution merging of color and texture information. Results obtained from testing our algorithm and comparisons to popular segmentation methods are provided

4

in chapter 4. Finally, chapter 5 provides the summary of our accomplishments in the creation of this new segmentation algorithm referred to as GSEG.

Chapter 2

Background

Years of research in the image and signal processing area have provided essential tools in the implementation of various applications. This chapter will introduce techniques that provide essential information required for the optimal implementation of our algorithm. We will discuss an edge-detection algorithm that provides the intensity of the edges present in the image. The edges are utilized to detect the individual regions in which the image is segmented and the direction in which the region growth will take place. From the statistical field, we bring an effective method to analyze grouped data; analyzing the grouped data is required to merge the regions that were over-segmented in the region growth procedure. And finally, a method for quantifying the quality of image segmentation techniques is introduced to evaluate the robustness of our algorithm.

2.1 Edge detection in a vector field

The initial seeds or clusters are created by detecting areas with no edges inside them; therefore, the first step of our algorithm is to employ an edge detection algorithm. Edge detection has been extensively studied in two dimensional space and was generalized for a multidimensional space by Lee and Cok [17]. Assuming that the image is a function $f(x, y)$, the edges can be defined as its first derivative $\nabla \mathbf{f} = [\partial f / \partial x; \partial f / \partial y]$. Since it is desired for the edges to be rotational invariant, the magnitude of the gradient is chosen. For

a vector field f, Lee and Cok expand the gradient vector to be defined as

$$
\mathbf{D}\left(\mathbf{x}\right) = \begin{bmatrix} D_1 f_1(\mathbf{x}) & D_2 f_1(\mathbf{x}) & \cdots & D_n f_1(\mathbf{x}) \\ D_1 f_2(\mathbf{x}) & D_2 f_2(\mathbf{x}) & \cdots & D_n f_2(\mathbf{x}) \\ \vdots & \vdots & \vdots & \vdots \\ D_1 f_m(\mathbf{x}) & D_2 f_m(\mathbf{x}) & \cdots & D_n f_m(\mathbf{x}) \end{bmatrix} \tag{2.1}
$$

where $D_j f_k$ is the first partial derivative of the kth component of f with respect to the jth component of $\mathbf{x}$. The distance from the point $\mathbf{x}$ with a unit vector $\mathbf{u}$ in the spatial domain $\mathbf{d} = \sqrt{\mathbf{u}^T \mathbf{D}^T \mathbf{D} \mathbf{u}}$ will be the corresponding distance traveled in the color domain. The vector which maximizes given distance is the eigenvector of the matrix $\mathbf{D}^T \mathbf{D}$ that corresponds to its largest eigenvalue.

In the special case of an RGB image, the gradient can be computed in the following manner: let u, v, w denote each color channel and x, y the spatial coordinates for a pixel. Defining the following variables to simplify the expression of the final solution:

$$
q = \left(\frac{du}{dx}\right)^2 + \left(\frac{dv}{dx}\right)^2 + \left(\frac{dw}{dx}\right)^2 \tag{2.2}
$$

$$
t = \left(\frac{du}{dx}\frac{du}{dy}\right) + \left(\frac{dv}{dx}\frac{dv}{dy}\right) + \left(\frac{dw}{dx}\frac{dw}{dy}\right) \tag{2.3}
$$

$$
h = \left(\frac{du}{dy}\right)^2 + \left(\frac{dv}{dy}\right)^2 + \left(\frac{dw}{dy}\right)^2 \tag{2.4}
$$

the matrix $\mathbf{D}^T \mathbf{D}$ becomes

$$
\mathbf{D}^T \mathbf{D} = \begin{bmatrix} q & t \\ t & h \end{bmatrix} \tag{2.5}
$$

and its largest eigenvalue λ is

$$
\lambda = \frac{1}{2}\left(q + h + \sqrt{(q+h)^2 - 4\left(qh - t^2\right)}\right) \tag{2.6}
$$

by calculating λ, the largest differentiation of colors is obtained, and the edges of the image can be defined as

$$
G = \sqrt{\lambda} \tag{2.7}
$$

7

2.2 One-Way Variance Analysis

One-way variance analysis allow us to take regions that have been separated due to occlusion, or small texture differences and merge them together. The core of one-way variance lies at highlighting the differences between groups that display multiple variables to investigate the possibility that multiple groups are associated with a single factor [18].

We considered the general case in which p variables $x_1, x_2, \ldots, x_p$ are measured on each individual group, and any direction in the p-dimensional sample of the groups is specified by the p-tuple $(a_1, a_2, \ldots, a_p)$. We can convert each multivariate observation $\mathbf{x}'_i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ into a univariate observation $y_i = \mathbf{a}'\mathbf{x}_i$ where $\mathbf{a}' = (a_1, a_2, \ldots, a_p)$. Since the samples are divided into g separate groups, it is useful to relabel each element using the notation y_{ij}, where i refers to the group that the element belongs to, and j is the location of the element on the ith group.

The objective of one-way variance is to find the optimal coefficients of the vector $\mathbf{a}$ which will yield the largest differences across groups and minimize the distances of elements within the group. To do this, the between-groups sum-of-squares and products matrix $\mathbf{B}_0$ and the within-groups sum-of-squares and products matrix $\mathbf{W}_0$ are define by

$$\mathbf{B}_0 = \sum_{i=1}^{g} n_i \left(\bar{\mathbf{x}}_i - \bar{\mathbf{x}} \right) \left(\bar{\mathbf{x}}_i - \bar{\mathbf{x}} \right)' \tag{2.8}$$

and

$$\mathbf{W}_0 = \sum_{i=1}^{g} \sum_{j=1}^{n_i} \left(\mathbf{x}_{ij} - \bar{\mathbf{x}}_i \right) \left(\mathbf{x}_{ij} - \bar{\mathbf{x}}_i \right)' \tag{2.9}$$

where the labeling $\mathbf{x}_{ij}$ is analogous to that of y_{ij}. $\bar{\mathbf{x}}_i = \frac{1}{n_i} \sum_{j=1}^{n_i} \mathbf{x}_{ij}$, the sample mean vector in the ith group, and $\bar{\mathbf{x}} = \frac{1}{n} \sum_{i=1}^{g} \sum_{j=1}^{n_i} \mathbf{x}_{ij} = \frac{1}{n} \sum_{i=1}^{g} n_i \bar{\mathbf{x}}_i$ is the overall sample mean vector. Since $y_{ij} = \mathbf{a}'\mathbf{x}_{ij}$, it can be verified that the sum of between-groups and within-groups become

$$SSB\left(\mathbf{a}\right) = \mathbf{a}'\mathbf{B}_0\mathbf{a} \ and \ SSW\left(\mathbf{a}\right) = \mathbf{a}'\mathbf{W}_0\mathbf{a} \tag{2.10}$$

With n sample members and g groups, there are $(g-1)$ and $(n-g)$ degrees of freedom between and within groups respectively. A test of the null hypothesis that there are no

differences in mean value among the g groups is obtained from the mean square ratio

$$F = \left\{ \frac{1}{(g-1)} \mathbf{a}'\mathbf{B}_0\mathbf{a} \right\} / \left\{ \frac{1}{(n-g)} \mathbf{a}'\mathbf{W}_0\mathbf{a} \right\} = \frac{\mathbf{a}'\mathbf{B}\mathbf{a}}{\mathbf{a}'\mathbf{W}\mathbf{a}} \tag{2.11}$$

where $\mathbf{B} = \frac{1}{(g-1)}\mathbf{B}_0$ is the between-group covariance matrix and $\mathbf{W} = \frac{1}{(n-g)}\mathbf{W}_0$ is the within-groups covariance matrix. Maximizing F with respect to $\mathbf{a}$ is done by differentiating F and setting the result to zero, yielding $\mathbf{B}\mathbf{a} - \left(\frac{\mathbf{a}'\mathbf{B}\mathbf{a}}{\mathbf{a}'\mathbf{W}\mathbf{a}} \right) \mathbf{W}\mathbf{a} = 0$. But at the maximum of F, $\frac{\mathbf{a}'\mathbf{B}\mathbf{a}}{\mathbf{a}'\mathbf{W}\mathbf{a}}$ must be a constant l, so the required value of $\mathbf{a}$ must satisfy

$$(\mathbf{B} - l\mathbf{W}) \, \mathbf{a} = 0 \tag{2.12}$$

This equation can be written $\left(\mathbf{W}^{-1}\mathbf{B} - l\mathbf{I} \right) \mathbf{a} = 0$, so l must be an eigenvalue, and $\mathbf{a}$ must be the eigenvector corresponding to the largest eigenvalue of $\mathbf{W}^{-1}\mathbf{B}$. This result will provide the direction in the p-dimensional data space, which will tend to keep the distances between each class small and simultaneously maintain the distances between classes as large as possible.

In the case where g is large, or if the original dimensionality is large, a single direction will provide a gross oversimplification of the true multivariate configuration. The term in (2.12) will generally possess more than one eigenvalue/eigenvector pair which can be used to generate multiple differentiating directions. Suppose that $\lambda_1 > \lambda_2 > \ldots \lambda_s > 0$ are the eigenvalues associated to the eigenvectors $\mathbf{a}_1, \mathbf{a}_2, \ldots, \mathbf{a}_s$. If we define new variates $y_1, y_2, \ldots$ by $y_i = \mathbf{a}_i'\mathbf{x}$, then the y_i are termed *canonical variates*.

The eigenvalues λ_i and eigenvectors $\mathbf{a}_i$ are gathered together so that $\mathbf{a}_i$ is the ith column of a $(p \times s)$ matrix $\mathbf{A}$, while λ_i is the ith diagonal element of the $(s \times s)$ diagonal matrix $\mathbf{L}$. Then, in matrix terms, equation (2.12) may be written as $\mathbf{B}\mathbf{A} = \mathbf{W}\mathbf{A}\mathbf{L}$, and the collection of canonical variates is given by $\mathbf{y} = \mathbf{A}'\mathbf{x}$. The space of all vectors $\mathbf{y}$ is termed the *canonical variate space*. In this space, the mean of the ith group of individuals is $\bar{\mathbf{y}}_i = \mathbf{A}'\bar{\mathbf{x}}_i$.

The Mahalanobis-squared distance between the ith and j^{th} group is given by

$$D^2 = (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)' \, \mathbf{W}^{-1} \, (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j) \tag{2.13}$$

9

and compared to the Euclidean distance of the group means in the canonical variate space, and substituting for $\bar{\mathbf{y}}_i$ and $\bar{\mathbf{y}}_j$ we obtain

$$
\begin{aligned}
d^2 &= (\bar{\mathbf{y}}_i - \bar{\mathbf{y}}_j)'(\bar{\mathbf{y}}_i - \bar{\mathbf{y}}_j) \\
&= (\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)'\,\mathbf{A}\mathbf{A}'\,(\bar{\mathbf{x}}_i - \bar{\mathbf{x}}_j)
\end{aligned}
\tag{2.14}
$$

but it can be proven that $\mathbf{A}\mathbf{A}' \equiv \mathbf{W}^{-1}$, thus substituting for $\mathbf{A}\mathbf{A}'$ above yields (2.13). Hence, by constructing the canonical variate space in the way described, the Euclidean distance between the group means is equivalent to the Mahalanobis distance of the original space.

Obtaining the Mahalanobis distance between groups is important, because it accounts for the covariance between variables as well as differential variances, and it is now the preferred measure of distance between two multivariate populations.

2.3   Evaluation of Image Segmentation Algorithms

To objectively measure the quality of our segmentation algorithm we have implemented a recently proposed measure of similarity, referred to as the Normalized Probabilistic Rand (NPR) index [19]. This method compares segmentations obtained from the tested segmentation algorithms and compares them to a set of manual segmentations available for the given image. The need of multiple manual segmentations for a single image is that there is not a singular correct segmentation; therefore, the set of multiple perspectives of correct segmentations becomes the ground-truth segmentations for the given image.

As its name implies, the NPR is a normalization of the Probabilistic Rand (PR) index. The PR index allows comparison of a test segmentation to a set of multiple ground-truth segmentation images through a soft nonuniform weighting of pixels pairs as a function of the variability in the ground truth set [20]. Assume that the ground-truth set is defined as $\{S_1, S_2, \ldots, S_K\}$ of an image $X = \{x_1, x_2, \ldots, x_N\}$ consisting of N pixels. Let S_{test} be the segmentation that is to be compared with the manually labeled set. We denote the label of pixel x_n as $l_n^{S_{test}}$ in the test segmentation and as $l_n^{S_k}$ in the kth manual segmentation.

10

The PR models label relationships for each pixel pair, where each human segmenter provides information $(c_{i,j})$ about each pair of pixels (x_i, x_j) as to whether the pair belongs to the same group 1 or belongs to different groups 0. The set of all perceptually correct segmentations defines a Bernoulli distribution for the pixel pair, giving a random variable with expected value denoted as $p_{i,j}$. The set $\{p_{ij}\}$ for all unordered pairs (i, j) defines the generative model of correct segmentations for the image X.

The Probabilistic Rand index is then defined as:

$$PR\left(S_{test}, \{S_k\}\right) = \frac{1}{\binom{N}{2}} \sum_{\substack{i,j \\ i<j}} \left[p_{ij}^{c_{ij}} \left(1 - p_{ij}\right)^{1-c_{ij}} \right] \tag{2.15}$$

This measure takes values between 0 and 1, where 0 means S_{test}, $\{S_1, S_2, \ldots, S_K\}$ have no similarities, and 1 means all segmentations are equal. Although the summation in (2.15) is overall possible pairs of N pixels, Unnikrishnan et al shows that the computational complexity of the PR index is $O\left(KN + \sum_k L_k\right)$.

The NPR index establishes a comparison method which meets the following requirements for comparison correctness:

1. Images whose ground-truth segmentations are not well defined cannot provide abnormally large values of similarity.

2. The comparison does not assume equal labeling of groups or same number of groups.

3. Boundaries that are well defined by the human manual segmentation are given greater importance than those regions that contain ill-defined boundaries.

4. Scores provide meaningful differences between segmentations of different images and different segmentations of the same image.

The NPR is defined as

$$NPR = \frac{PR - \mathbb{E}\left[PR\right]}{Max\left[PR\right] - \mathbb{E}\left[PR\right]} \tag{2.16}$$

11

where the maximum possible value of the PR to be 1, and the expected value of the PR index is computed as

$$\mathbb{E}\Big[PR(S_{test}, \{S_k\})\Big] = \frac{1}{\binom{N}{2}} \sum_{\substack{i,j \\ i<j}} \Big\{ \mathbb{E}\Big[\mathbb{I}(l_i^{S_{test}} = l_j^{S_{test}})\Big] p_{ij}$$
$$+ \mathbb{E}\Big[\mathbb{I}(l_i^{S_{test}} \neq l_j^{S_{test}})\Big](1 - p_{ij}) \Big\} \qquad (2.17)$$
$$= \frac{1}{\binom{N}{2}} \sum_{\substack{i,j \\ i<j}} \Big[p'_{ij} p_{ij} + (1 - p'_{ij})(1 - p_{ij}) \Big]$$

where $\mathbb{I}$ is the identity function, and $p'_{ij} = \mathbb{E}[\mathbb{I}(l_i^{S_{test}} = l_j^{S_{test}})]$ is defined as the weighted sum of $PR(S_k^\phi, \{S_k\})$.

Let Φ be the number of images in a data set and K_ϕ the number of ground-truth segmentations of image ϕ. Then, p'_{ij} can be expressed as:

$$p'_{ij} = \frac{1}{\Phi} \sum_\phi \frac{1}{K_\phi} \sum_{k=1}^{K_\phi} \mathbb{I}\Big(l_i^{S_k^\phi} = l_j^{S_k^\phi}\Big) \qquad (2.18)$$

The normalization of the Probabilistic Rand is important, since it provides a meaningful value of similarity when segmentations of the same image are compared and a low value of similarity when segmentations of different images are compared.

Chapter 3

Proposed Algorithm

An overview of the technique developed to segment images is provided here to serve as a reference to the reader. The algorithm is composed of three different modules. The first module uses an edge-detection algorithm to dynamically create regions composed of contiguous pixels that display similar gradient values. The second module consists of creating an additional channel from the image. This channel contains the texture information of the image. The last module combines the texture information and the initial segmentation map obtained in the first module to merge regions that display the same variance of colors and texture. The following sections will describe the detailed technique for each of the three modules.

3.1 Region Growth and Dynamic Seed

The quality of region-growing techniques is highly dependent on the initial locations chosen to initialize the growing procedure. We propose an alternative process for region growth that does not depend exclusively on the initial assignment of clusters for the final segmentation. The procedure searches for regions in the image where no edges have been detected. The selected regions form the initial set of clusters to segment the image. Sky, skin, and in general, objects of no strong color variance are selected in this step. Subsequent clusters are incorporated with various levels of edge density during the growth procedure, to account for all other objects that are found in natural images. A flowchart for this module
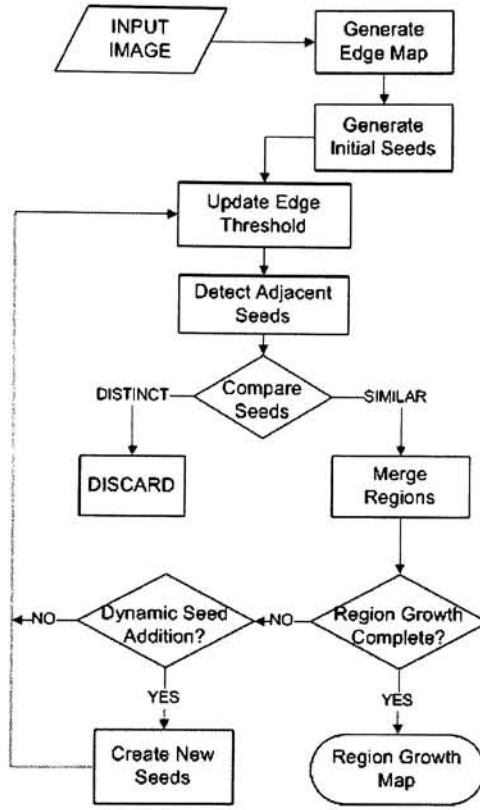
Figure 3.1: Flowchart of Region Growth Procedure

is shown in Fig. 3.1 and the detailed explanation of each step is described below.

3.1.1 Initial Seed Generation

Using the magnitude of the gradient $G(i, j)$ of the color image field, computed as described in 2.1, the edge map of images is obtained. Ideally, a threshold value could be selected to provide us with the most edges, while ignoring noise present in images. The problem is that the nature of images does not allow for this disposition. A threshold that may correctly delineate the boundary of a given region may allow other regions to be merged. Due to this factor, we initiate our algorithm by selecting regions that do not present any edges within, and, if such regions are not found, the edge value or threshold is increased until regions are detected. To prevent multiple seed generation within homogeneous and

14

connected regions, the region selection at this stage is restricted to clusters of pixels which are larger than 0.5% of the image. Each individual cluster is assigned a particular label for differentiating purposes. This label map is referred as the Parent Seeds (PS). The labeling procedure uses the general procedure outlined in reference [21].

1. Run-length encode the input image.

2. Scan the runs, assigning preliminary labels and recording label equivalences in a local equivalence table.

3. Resolve the equivalence classes.

4. Relabel the runs based on the resolved equivalence classes.

3.1.2 Region Growth

The procedure continues by increasing the threshold found in the initial seed generation and detecting new regions or child seeds that fall below the new threshold. These child seeds need to be classified into adjacent-to-existent or non-adjacent seeds. In order to make the region growth process efficient, it is important to also know the parent seed to which the child is adjacent. The objective is to be able to process all adjacent child seeds in a vectorized approach. To achieve this task we proceed to first detect the outside edges of the PS map using a nonlinear spatial filter. The filter operates on the pixels of a 3×3 neighborhood, and the response of its operation is assigned at the center pixel of the neighborhood. The size of neighborhoods being operated will be assumed to be 3×3 unless specified otherwise. The filter operates according to

$$F(i,j) = \begin{cases} 0 & \text{if } PS(i,j) > 0, \\ 0 & \text{if } \sum_{(m,n)\in\beta} PS(m,n) \text{ is } 0, \\ 1 & \text{otherwise} \end{cases} \tag{3.1}$$

where β is the neighborhood being operated. The result of applying this filter is a mask indicating the borders of the PS map.

The child seeds are individually labeled, and the ones adjacent to the parent seeds are identified by performing an element-by-element multiplication of the parent seeds edge mask and the labeled child map. The remaining pixels are referred to as the *adjacent child pixels*. The pixels whose labels are members of the set of labels remaining after the multiplication become part of the *adjacent child seeds* map. For the proper addition of adjacent child seeds, it is necessary to compare their individual color differences to their parents to assure a homogeneous segmentation. Reduction of the number of seeds to be evaluated is accomplished by attaching to their parents the child seeds that have a size smaller to the minimum seed size (MSS). In our algorithm, the MSS is set to 0.01% of the image.

The child seed sizes are computed utilizing sparse matrix storage techniques to allow for the creation of large matrices with low memory costs. Sparse matrices store only the nonzero elements of the matrix, together with their location in the sparse matrix (indices). The size of each child seed is computed by creating a matrix of $M \times N$ columns by C rows, where M is the number of columns of pixels in the image, N the number of rows, and C the number of adjacent child seeds. The matrix is created by allocating a 1 at each column in the row that matches the pixel label. The pixels that do not have a label are ignored. By summing all the elements along each row, we obtain the number of pixels per child seed. This procedure is useful for any operation that requires the knowledge of the number of elements per group in the segmentation algorithm.

To attach regions an association between child seeds and their parents is required. The adjacent child pixels provide the child labels, but not the parent labels. A new spatial filter is applied to the PS map to obtain the parent labels. The filter response at each center point is equal to the maximum pixel value in its neighborhood. The association between child and parent can now be obtained by creating a matrix with the first column composed of the adjacent child pixels and the second column, with labels found at the location of the

adjacent child pixels in the matrix obtained after applying the maximum value filter to the PS map. It is important to note that non-linear filters are used to provide information about the seeds and not to directly manipulate the image; therefore, it does not impair the final result.

The functionality of the association matrix is manifold. It provides the number of child pixels that are attached to parent seeds and also identifies which child seeds share edges with more than one parent seed. Child seeds smaller than MSS can now be directly attached to their parents. Child seeds that share less than 5 pixels with their parents and are larger than MSS are returned to the unsegmented region to be processed when the region shares a more significant border. Finally, the remaining seeds are compared to their parents to analyze if they should be added to a parent seed or not.

Given that spatial regions in images vary gradually, only the nearby area of adjacency between parent and child is compared to provide a true representation of the color difference. This objective is achieved by using two masks that will exclude the areas of both parent and child seeds that are distant from their common boundaries. The first mask is a dilation of the PS map using an octagonal structuring element with a distance of 15 pixels between the center pixel to the sides of the octagon, as measured along the horizontal and vertical axis. The second mask is the same dilation but this time applied to the *adjacent child seeds* map. The two masks mutually exclude the pixels that fall beyond each other's dilation masks. The values used in our algorithm are optimized to work with images that range from 300×300 pixels of resolution to 1000×1000 pixels.

The comparison of regions is performed using the Euclidian distance between the mean colors of the groups. The reason for choosing this method is that only the nearby area of regions is being compared, and, therefore, the increased complexity of using the Mahalanobis distance does not improve the results, because the variance of the regions compared will be small. Also, prior to comparing the colors, the image is converted to the CIE L*a*b* color space, assuring that comparing colors using the Euclidean distance is similar to the

differentiation of colors by the human eye. The maximum color distance to allow the integration of the child seed to the parent seed is set to 20. This distance is chosen to allow the differentiation of at least 10 different colors along the range of the a* channel or b* channel.

3.1.3 Dynamic Seed Generation

When the parent seeds are created in the initial seed generation, the regions represented by these seeds are characterized by areas of the image that have no texture. These areas can be instances of sky, water, skin, and in general, regions where there is either no color variance or a gradual transition from one color to the next. The dynamic addition of seeds to the PS map is our answer to include the remaining regions that display different levels of edge intensities through them but are part of the same identifiable object. Dynamic seed generation consist of selecting a set of threshold values at which additional seeds are added to the parent seeds. The threshold values are adjusted to account for the exponential decay of edge values as seen in Fig. 3.2. Ranges in the low-edge values account for large areas in the image. To incorporate new areas, the threshold values need to increment exponentially to include additional elements of considerable size into the segmentation map. The threshold values selected for the addition of new seeds are: 15, 20, 30, 50, 85, and 120 accounting for an increment of $\pm10\%$ of the area of the image added at each interval.

At these intervals, the addition of new seeds follow a similar procedure to the method explained in 3.1.2. The regions that fall below the selected edge threshold are detected. All the regions that are not attached to any parent seeds and are larger than the MSS are added to the PS map. For the addition of new seeds that share borders with existent seeds, they are required to meet two qualifications: 1) the group must be large enough to become a group by itself, and 2) the color differences between the region and its neighbor must be greater than the maximum color difference allowed.

(a) Original   (b) Edge Map
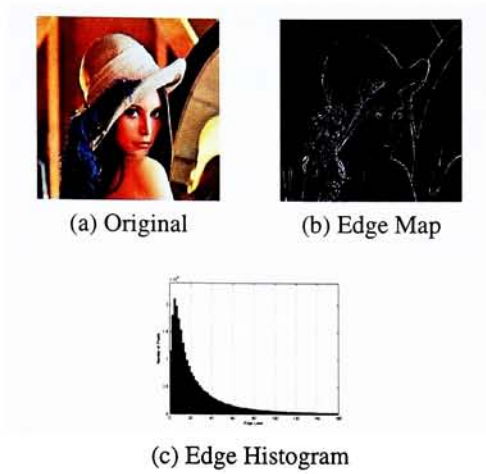
(c) Edge Histogram

Figure 3.2: Histogram of Edge Values for Lena

3.1.4 Seed Growth Tracking and Classification

Region growth without feedback of the growth rate of each seed may cause parent seeds to overflow into regions of similar colors but different textures. Regions in images display similar edge density throughout the region; therefore, to maintain homogeneity, the regions that are created at low gradient levels and slow their growth rate should be classified as a grown seed and should be removed from the growth procedure. The size tracking of each seed is performed at each dynamic seed addition interval. The number of pixels per seed is computed at each interval, and when the increment of a parent seed does not reach above 5% of its original size, the growth of this seed is stopped. When the last interval has been reached, all the identifiable regions have been given a label and all remaining areas are edges of the segmented regions. At this stage all seeds are allowed to grow to complete the region growth procedure.

3.2 Texture Channel Generation

Much of the problem in image segmentation algorithms is caused by the presence of regions that contain distinct patterns. The issue is that patterns are composed of multiple

19

shades of colors and cause over-segmentation and misinterpretation of the edges surrounding the patterned object. These objects are referred to in the computer vision industry as textures. Texture regions may contain regular patterns such as a brick wall, to irregular patterns such as leopard skins, bushes and many objects found in nature. The presence of texture in images is so large and descriptive of objects that we have decided to generate an additional channel containing this important information.

A method for obtaining information of patterns within an image is to evaluate the randomness present at various areas of the image. Entropy provides a measure of uncertainty of a random variable [22]. If the random variable is composed from the pixel values of a region, the entropy will define the randomness associated to the region being evaluated. Texture regions contain various colors and shades; therefore, texture regions will contain a specific value of uncertainty associated with them, providing a structure to merge regions that display similar characteristics.

Information theory introduces entropy as the quantity which agrees with the intuitive notion of what a measure of information should be [22]. In accordance with this supposition, we can select a random group of pixels **s** from an image, with a set of possible values $\{a_1, a_2, \ldots, a_J\}$. The probability for a specific value a_j to occur is $P(a_j)$, and it contains

$$I(a_j) = log\frac{1}{P(a_j)} = -logP(a_j) \tag{3.2}$$

units of information. The quantity $I(a_j)$ is referred to as the self-information of a_j. If k values are present on the set, the law of large numbers stipulates that, for a sufficiently large value of k, symbol a_j will on average be output $kP(a_j)$ times. Thus the average self-information obtained from k outputs is

$$-kP(a_1)logP(a_1) - \ldots - kP(a_J)logP(a_J) \tag{3.3}$$

or

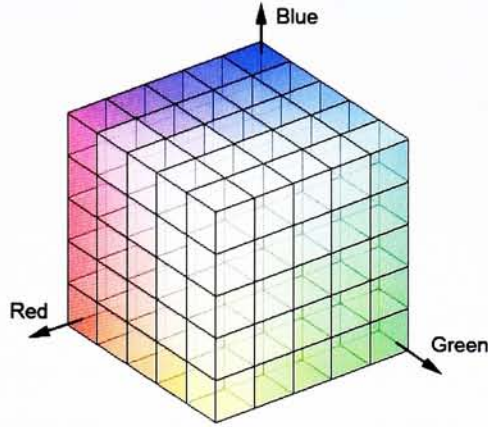$$-k\sum_{j=1}^{J}P(a_j)logP(a_j). \tag{3.4}$$

Figure 3.3: Quantization of Colors

The average information per source output, or entropy is defined by

$$H(\mathbf{s}) = -\sum_{j=1}^{J} P(a_j) log P(a_j) \qquad (3.5)$$

This quantity is defined for a single random variable, and for the case of color, multiple variables are used. To take advantage of the color information without extending the process to compute the joint entropy, the colors in an image are quantized into 6^3, or 216 different colors. The quantization of colors can be done using uniform quantization which cuts the RGB color cube into smaller boxes, and then maps all colors that fall within each box to the color at the center of that box. Uniform quantization is represented in Fig. 3.3.

After the colors have been quantized, each pixel of the image can be indexed to one of the 216 representative colors, effectively reducing the probability of each color occurring to a one-dimensional random variable. To create the texture channel, the local entropy is computed on a 9-by-9 neighborhood around each pixel in the indexed image, and the resulting value is assigned to the center pixel of the neighborhood.

3.3 Multiresolution Region Merging

The texture channel obtained from the second module is combined with the color information to describe the characteristics of each region segmented by the region growth module. Using a multivariate analysis of the independent regions, the resultant Mahalanobis distances between groups is used to merge similar regions.

To this point the segmentation of the image has been performed with an absence of information about the individual regions. Now that the image has been segmented into different groups, information can be gathered from each individual region. Given that we have four sources of information (Red, Green, Blue, and Texture) and individual regions displaying a different number of pixels per group, we require a suitable method to display the data in order to investigate relationships of the regions. The data can be modeled using an $(N * P)$ matrix, where N is the total number of pixels in the image, and P are the total number of variables that contain information about each pixel. G is the total number of groups in which the image has been segmented in the region growth procedure; then the matrix is composed of G separate sets. The objective is to obtain a mean value for each group that is used to compare the different groups. The method used to achieve this objective is a one-way analysis of variance, which is explained in detail in section 2.2.

The Mahalanobis-squared distances are obtained for each pair of groups from the one-way analysis of the data. The algorithm uses these distances to find the similar groups and merge them. Once a group has been merged, the similarity of this group to the others is unknown, but required if the new group needs to be merged to other similar groups. To prevent the need to re-evaluate the Mahalanobis distances for the groups after each region merging has occurred, an alternate approach was introduced.

Having the distances between groups, the smallest distance value is found. This value only provides one pair of groups; therefore, the similarity value is increased until a larger set of group pairs is obtained. Five group pairs were found to be an adequate number of groups to reduce computation time without merging groups inadequately. We initiate by merging the smaller group in this set and then continue to merge the next larger group.

After the first merge, a check is performed to see if one of the groups being merged is now part of a larger group. In this case all the pair combinations of the groups should belong to the pairs selected initially in the set to be merged together.

Once all the pairs of the set have been processed, the Mahalanobis distance is recomputed for the new segmentation map, and the process is repeated until a desired number of groups is achieved. The value of similarity obtained, after the desired number of groups has been achieved, should be used as a minimum value of merging to assure that all the images display a similar level of segmentation. A flowchart of the procedure is shown in Fig. 3.4.
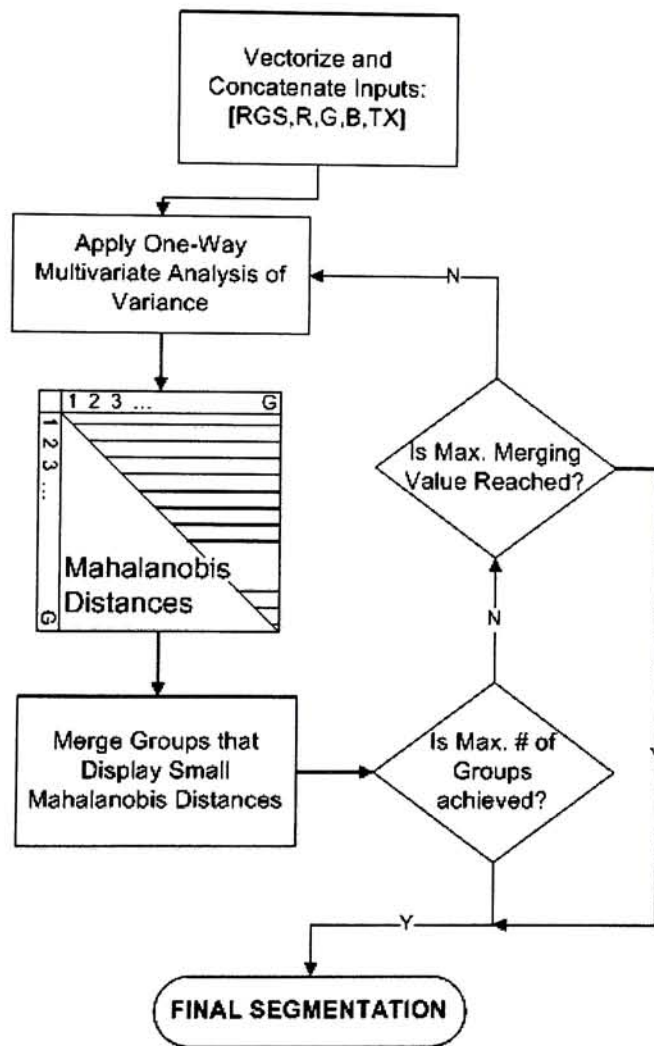
Figure 3.4: Flowchart of Merging Procedure

Chapter 4

Results

Until recently, different segmentation algorithms proved their effectiveness by displaying the results obtained on a limited set of images. Hebert et al introduced a method for actually assigning a quantitative value to the quality of a given segmentation by calculating the Normalized Probabilistic Rand (NPR) index [19]. The technique involved in the calculation of the NPR index is summarized in section 2.3. Because the NPR provides a value which is directly related to the manual segmentations utilized in the evaluation process, a set of manual segmentations is required that display the following characteristics:

1. It cannot be chosen selectively to favor a given algorithm.

2. It displays various scenarios with multiple levels of complexity.

3. It contain more than one individual perspective.

4. It can be accessed by anyone to perform the same test on diverse algorithms.

Table 4.1: Segmentation algorithm results

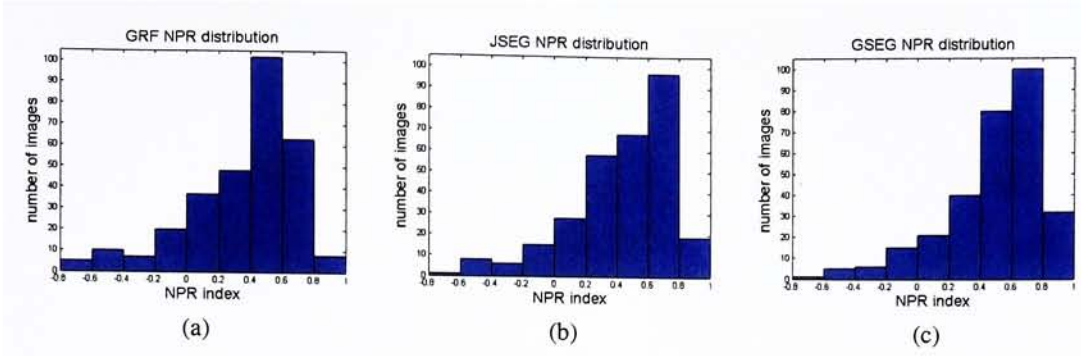|	GRF	JSEG	GSEG
Avg. Time (sec)	280	28	46
Avg. NPR	0.358	0.440	0.486
Std. Dev. NPR	0.345	0.319	0.313
Environment	C	C	MATLAB

Figure 4.1: Distribution NPR scores. (a)GRF; (b)JSEG; (c)GSEG

Such a set is available on the publicly accessible Berkeley Segmentation Database. This database provides 1633 manual segmentations for 300 images created by 30 human subjects [23]. State-of-the-art algorithms were chosen to compare the quality of the segmentation results. These segmentation techniques are Fusion of Color and Edge Information for improved Segmentation and Edge Linking (GRF) [6], the unsupervised segmentation of color-texture regions in images and video (JSEG) [16], and the novel algorithm automatic segmentation by dynamic region growth and multiresolution merging (GSEG). To prevent any discrepancy at the time of comparing the results, all the available images were segmented using the available segmentation algorithms on the same machine. The testing computer has a Pentium 4 CPU 3.20GHz, and 1.00 GB of RAM. The GRF and JSEG algorithms were run from the executable file provided by Rochester Institute of Technology and the University of California respectively. The proposed method was processed using MATLAB R2006a.

The normalization factor was computed by evaluating the Probabilistic Rand (PR) for all available manual segmentations, and the expected index obtained was 0.6064. Results obtained for the distinct methods are displayed in table 4.1. The results show that the GSEG algorithm has the highest overall NPR results, and the variance of the results has the narrowest spread, proving that the proposed algorithm performs consistently better than the other algorithms. Fig. 4.1 display the distribution of the NPR scores for all the tested images. It can be observed that 273 out of 300 images using the proposed algorithm are
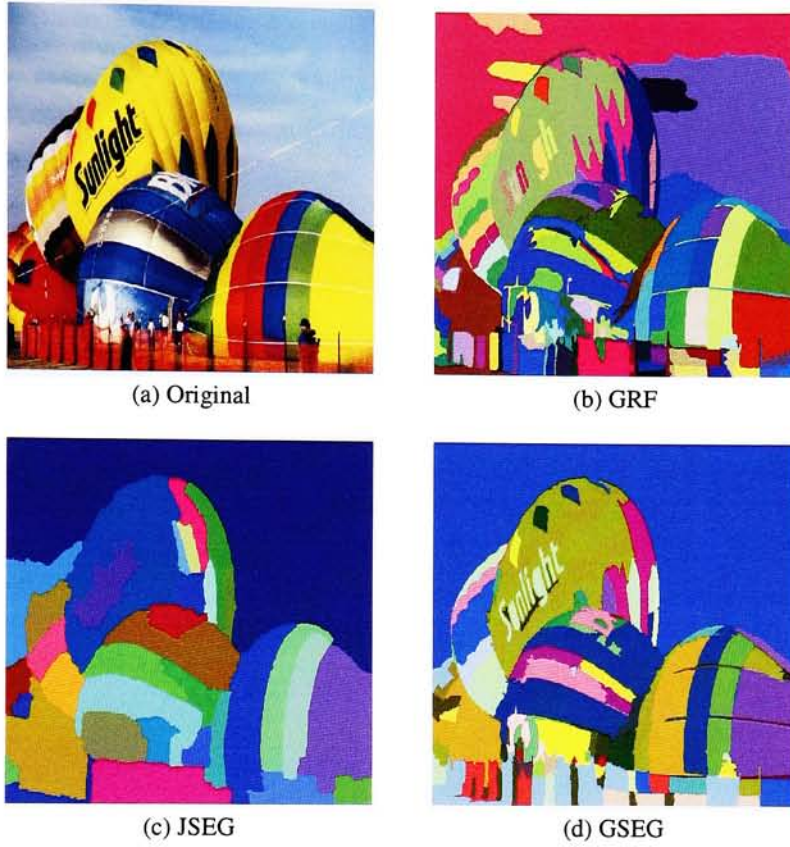
(a) Original

(b) GRF

(c) JSEG

(d) GSEG

Figure 4.2: Balloon Results

distributed into the top half or acceptable segmentation range. This is similar to the number of segmentations that fall within this range using the JSEG algorithm. The actual improvement can be seen in the number of segmentation scores that fall within the range of very good segmentation results $[0.7 < NPR < 1]$. These numbers for the GRF, JSEG, and GSEG were 47, 74, and 98 respectively. This indicates that close to a third of the images segmented using our algorithm match closely to the segmentations performed by human beings. The evaluation of the segmentation algorithms were performed by maintaining the standard parameters available as the inputs for the segmentation programs.

Some of the results obtained from running the GSEG algorithm, as well as the other segmentations methods without any parameter tuning are shown in Figs. 4.2 - 4.7. Clear
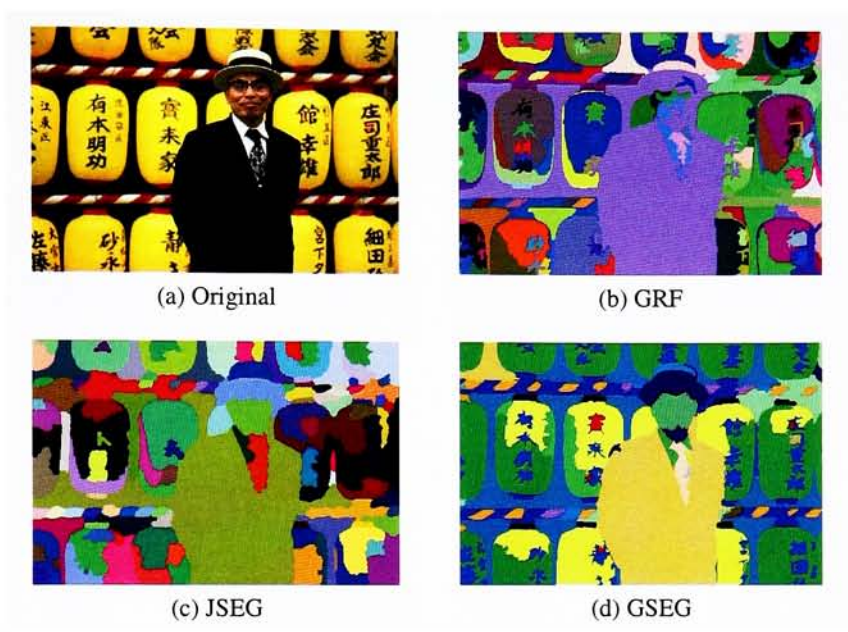
(a) Original

(b) GRF

(c) JSEG

(d) GSEG

Figure 4.3: China Results

(a) Original

(b) GRF

(c) JSEG

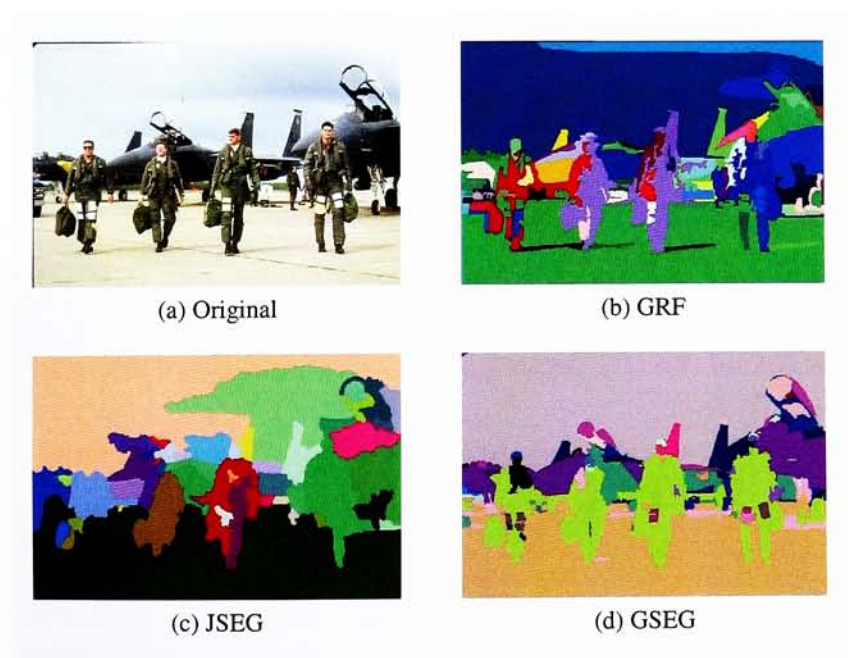(d) GSEG

Figure 4.4: Pilot Results

28

(a) Original   (b) GRF   (c) JSEG   (d) GSEG

Figure 4.5: London Results

(a) Original   (b) GRF   (c) JSEG   (d) GSEG

Figure 4.6: Tribal Results

(a) Original   (b) GRF   (c) JSEG   (d) GSEG

Figure 4.7: Lady Results

29

advantages can be seen on the level of detail achieved from our segmentation in comparison to the JSEG segmentation for all the images. The boundaries of the GRF procedure match directly the boundaries of the object in the image. Fig. 4.4(c), and Fig. 4.5(c) displays the disadvantage of guiding the segmentation method based on the quantization of colors to create the initial clusters. In these images the sky has been oversegmented, because the change of light provides different shades of blue. Advantages of the multiresolution merging method can be seen on Figs. 4.3(d) and 4.7(d). In these results multiple regions that create a pattern are assigned to the same class, allowing the algorithm to reduce the total number of classes without losing information obtained from multiple similar regions that are not adjacent to each other. The GRF results have the same level of detail as the GSEG approach, but its lack of texture modeling does not allow it to differentiate objects with similar colors but different texture. Figs. 4.4(b), 4.5(b), and 4.6(b) have regions merged that display similar colors but contain substantially different textures.

Chapter 5

Conclusion

This work provides an effective method for automatic image segmentation on simple to complex images. The algorithm is based on color edge-detection and dynamic region growing, completed by a multiresolution region merging. The segmentation procedure has been tested on the publicly available Berkeley database, and the quality of its results has been measured. The robustness of our algorithm is displayed on the results, along with those obtained on the same image when segmented by other methods. Future research would be focused on object classification based on this segmentation algorithm.

References

[1] J. B. McQueen, "Some methods for classification and analysis of multivariate observations," in *Proc. of 5th Berkeley Symp. on Mathematical Statistics and Probability*, vol. 1, pp. 281–296, 1967.

[2] J. Wu, H. Yan, and A. N. Chalmers, "Color image segmentation using fuzzy clustering and suppervised learning," *Journal of Electronic Imaging*, pp. 397–403, October 1994.

[3] P. Schmid, "Segmentation of digitized dermatoscopic images by two-dimensional color clustering," *IEEE Transactions on Medical Imaging*, vol. MI-18, pp. 164–171, February 1999.

[4] T. N. Pappas, "An adaptive clustering algorithm for image segmentation," *IEEE Transactions on Signal Processing*, vol. 40, pp. 901–914, April 1992.

[5] M. M. Chang, M. I. Sezan, and A. M. Tekalp, "Adaptive bayesian segmentation of color images," *Journal of Electronic Imaging*, vol. 3, pp. 404–414, October 1994.

[6] E. Saber, A. M. Tekalp, and G. Bozdagi, "Fusion of color and edge information for improved segmentation and edge linking," *Image and Vision Computing*, vol. 15, pp. 769–780, 1997.

[7] J. Fan, D. K. Y. Yau, A. K. Elmagarmid, and W. G. Aref, "Automatic image segmentation by integrating color-edge extraction and seeded region growing," *IEEE Transactions on Image Processing*, vol. 10, pp. 1454–1466, October 2001.

[8] S. Y. Wanand and W. E. Higgins, "Symmetric region growing," *IEEE Transactions on Image Processing*, vol. 12, pp. 1007–1015, September 2003.

[9] L. Liu and S. Sclaroff, "Region segmentation via deformable model-guided split and merge," in *International Conference on Computer Vision*, vol. 1, (Vancouver, BC, Canada), pp. 98–104, 2001.

[10] C. D'Elia, G. Poggi, and G. Scarpa, "A tree-structured markov random field model for bayesian image segmentation," *IEEE Transactions on Image Processing*, vol. 12, pp. 1259–1273, October 2003.

[11] H. Derin and H. Elliott, "Modeling and segmentation of noisy and textured images using gibbs random fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-9, pp. 39–55, January 1987.

[12] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 674–693, July 1989.

[13] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Transactions on Image Processing*, vol. 4, pp. 1549–1560, November 1995.

[14] G. Fan and X.-G. Xia, "Wavelet-based texture analysis and synthesis using hidden markov models," *IEEE Transactions on Circuits and Systems*, vol. 50, pp. 106–120, January 2003.

[15] J. Chen, T. N. Pappas, A. Mojsilovic, and B. Rogowitz, "Adaptive image segmentation based on color and texture," in *International Conference on Image Processing*, vol. 3, pp. 777–780, june 2002.

[16] Y. Deng and B. S. Manjunath, "Unsupervised segmentation of color-texture regions in images and video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, pp. 800–810, August 2001.

[17] H.-C. Lee and D. R. Cok, "Detecting boundaries in a vector field," *IEEE Transactions on Signal Processing*, vol. 39, pp. 1181–1194, May 1991.

[18] W. J. Krzanowski, *Principles of Multivariate Analysis*, ch. 11. Oxford University Press, 1988.

[19] R. Unnikrishnan, C. Pantofaru, and M. Hebert, "Toward objective evaluation of image segmentation algorithms." Accepted for future publication in IEEE Transactions on Pattern Analysis and Machine Intelligence.

[20] R. Unnikrishnan and M. Hebert, "Measures of similarity," in *IEEE Proceedings Workshop Computer Vision Apllications*, 2005.

[21] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, vol. 1, pp. 28–48. Addison-Wesley, 1992.

[22] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. Wiley Interscience, 1991.

[23] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *IEEE International Conference on Computer Vision*, vol. 2, (Vancouver, BC, Canada), pp. 416–423, 2001.