NATURAL SCENE SEGMENTATION BASED ON INFORMATION FUSION AND HIERARCHICAL SELF-ORGANIZING MAPS

by

Manasi Datar

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Computer Science

Approved:

Heng-Da Cheng Major Professor	Xiaojun Qi Committee Member
Gregory W. Jones Committee Member	Laurens H. Smith, Jr. Interim Dean of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2005

This thesis is dedicated to my parents, for having faith in me. Their love, support and encouragement, even from across the world, has been a never-ending source of inspiration.

Aai, Baba... I made it!

ABSTRACT

Natural Scene Segmentation Based on Information Fusion
and Hierarchical Self-Organizing Maps

by

Manasi Datar, Master of Science
Utah State University, 2005

Major Professor: Heng-Da Cheng
Department: Computer Science

This thesis focuses on an intuitive approach to natural scene segmentation. This research uses color and texture features in cooperation to provide comprehensive knowledge about every pixel in the image. A novel scheme for the collection of training samples, based on the notion of homogeneity, is proposed. Natural scene segmentation is carried out using a two-stage hierarchical self-organizing map. The first stage of the network employs a fixed-sized two-dimensional map that captures the dominant color and texture features of an image, in an unsupervised mode. The second stage combines a fixed-sized one-dimensional feature map and color merging, to control the number of color clusters formed as a result of the segmentation. The proposed method confirms that the self-learning ability and adaptability of the self-organizing map, coupled with the information fusion mechanism of the hierarchical network, leads to good segmentation results. These are further confirmed by extensive tests on a variety of natural scene images.

(62 pages)

ACKNOWLEDGMENTS

I would first like to thank my advisor and mentor, Dr. Heng-Da Cheng, for his support, encouragement, and inspiration, not only in my research work, but also throughout my years as a masters student. He has been a guiding light in all my endeavors, and I will forever be grateful to him.

I also express heartfelt thanks to Dr. Xiaojun Qi and Dr. Greg Jones for their patience and valuable advice. This thesis would not have taken shape without their valuable comments.

Friends are the *elixir* of life. I am grateful to all my friends, especially Anisha, Mihir and Brian, for their constant encouragement and support. Thank you. Life without all of you would have been quite tough.

Last, but definitely not the least, I would like to thank my parents and my dear sister, Ila. Their unconditional love and support has made me the person I am today.

Manasi Datar

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Segmentation

Splitting an image into several components by assigning one of these components to each pixel in the image is termed image segmentation. A good segmentation may be recognized from the characteristics of its output components: each component should be spatially cohesive as well as spatially accurate, while different components should be dissimilar [16]. A formal definition of image segmentation is as follows [40]: If P() is a homogeneity predicate defined on groups of connected pixels, then segmentation is a partition of the set F into connected subsets or regions $(S_1, S_2, ..., S_n)$ such that

$$\bigcup_{i=1}^{n} S_i = F, \quad with\ S_i \bigcap S_j = \Phi \quad (i \neq j)$$

The uniformity predicate $P(S_i)$ = true for all regions, S_i, and $P(S_i \bigcup S_j)$ = false, when $i \neq j$ and S_i and S_j are neighbors.

Color image segmentation attracts more and more attention, due to the following reasons: 1) color images can provide more information than gray level images; 2) the power of personal computers is increasing rapidly, and PCs can be used to process color images [6]. Color image segmentation approaches can be broadly classified in:

- edge and line oriented segmentation,
- region growing methods, and
- clustering.

Edge and line oriented segmentation approaches work on image data either through individually analyzing each data band (e.g. the RGB channels) or through considering the whole vector space by using a gradient calculated in a two-dimensional vector space [11,30]. After edge extraction, a postprocessing stage is be applied to create the segments. Region growing and splitting methods deal commonly with feature extraction and thresholding.

Ohta [38] adopts the Karhunen-Love transformation to a region-splitting phase to extract a significant color feature set. Ohlander, Price, and Reddy [37] employ a similar approach that uses a significant threshold to split regions. The histogram is calculated for each color component, and the best peak is detected and adopted to split the regions. Clustering techniques for color images often deal with fuzzy approaches. A commonly employed method is clustering, using fuzzy c-means to detect different classes [21, 22, 31]. This method is restricted in that in most cases, the number of classes, must be known a priori. Efforts to combine region growing and clustering have not been entirely successful. For example, Liu and Yang [33] use scale space filter (SSF) and a Markov random field (MRF) model in their segmentation scheme. However, the SSF is histogram-based and cannot provide reliable a priori knowledge for the MRF model. Traven [46] has investigated the application of a competitive learning algorithm to statistical pattern classification using both local spectral and contextual features, but it is a supervised learning procedure in which the image must first be manually segmented.

An artificial neural network, based on the idea of preserving the topology of the original input data set, was first proposed by Kohonen [26], and is called the self-organizing map (SOM) network. Unlike simple competitive learning methods in which only the winning neuron is allowed to learn, the neurons in the neighborhood of the winning neuron also participate in the learning process, leading to an ordered feature-mapping that can be exploited in many applications (e.g. clustering in color images). Ghosal and Mehrotra [13] describe a Kohonen self-organizing feature map for segmenting range images using local information provided by the orthogonal Zernike moments. Papamarkos [41] proposed an approach for color reduction that uses both the image color components and local image characteristics to feed a 1D Kohonen self-organizing feature map. The limitation of this method is that the final number of colors has to be specified a priori. Lampinen and Oja [27] have proposed a multi-layer SOM, HSOM,[1] as an unsupervised clustering method. The HSOM has the following properties:

[1] Architecture of the hierarchical self-organizing map is explained in detail in Sec. 2.2.

- It forms arbitrarily complex clusters,
- It provides a natural measure for the distance of a point from a cluster by giving appropriate weights to all the points belonging to the cluster, and
- It produces clusters that better match the desired classes than the direct SOM or the classical k-means or ISODATA algorithms.

The HSOM is, thus, a conceptually simple strategy that is better than most competitive learning techniques.

1.2 Texture

Intuitively, texture refers to a pattern of closely placed elements in such a manner that the pattern somehow repeats itself. Texture has long been an active area of research in the image processing community. As early as 1979, Haralick provided the classic survey of texture measures [17]. In spite of this long history of research, perhaps because texture is such an intuitive concept, no definition of the term has been universally acknowledged. Although researchers have often noted the lack of a common definition, the literature in the field has long reached a consensus on what texture is, how it can be analyzed within an image, and when texture analysis is useful in image processing. Most commonly, texture analysis is employed when there is a significant variation between the intensities of adjacent pixels, even in the absence of an "edge" separating the pixels. It is recognized that there is a difference in the meaning of texture, depending on the nature of the images themselves. For example, it is recognized that natural textures tend to be random, but artificial textures tend to be regular and periodic [23].

Image texture analysis methods use different descriptors of texture. Each of these descriptors capture a different part of the intuitive understanding of texture. Suggested descriptors include hidden Markov models [20], Markov random fields [4], image moments [35], co-occurrence [17] and correlation matrices [5], and filtering methods [12,24,48]. In his survey mentioned above, Haralick lists and describes a number of texture extraction methods that he divides into two types: structural and statistical. The structural

approaches use primitives to describe texture elements and placement rules to describe the spatial relationship between elements. This approach is better suited to textures that exhibit a regular macro-structure and will not be discussed further. The statistical approach assumes that texture is characterized by the gray value pattern in a neighborhood surrounding the pixel [25]. Local coherence estimates [45], Gabor filter banks [47], amplitude envelopes of band-pass filters [1], and multiple components' frequency estimates [19] have also been used to describe texture features.

Laws [28, 29] investigated three texture feature generation methods in detail: co-occurrence, correlation, and spatial-statistical techniques and developed a coherent set of "texture energy" masks.[2] All the masks were derived from three simple one-dimensional nonrecursive filters. These may be convolved with each other to give a variety of one- and two-dimensional filters. Laws found a set of bandpass and highpass directional filters, implemented as 5x5 masks, to be the most useful. The outputs from these masks are passed to "texture energy" filters. Texture energy images are used either directly or via principal components analysis as feature images for segmentation and/or classification. Laws used Brodatz [2] textures and other images to compare his masks with co-occurrence and correlation based features. He achieved pixel classification success rates of 94, 72, and 65 percent respectively. Pietikainen et al. [43, 44] tested Laws, co-occurrence contrast, and "edge per unit area" operators on Brodatz and geological terrain types. They found that the Laws operators performed consistently better than either edge- or co-occurrence-based features. Miller and Astley [34] used Laws' masks and morphological operators to detect glandular tissue in breast X-rays. They found that Laws' masks R5R5, L5L5, and S5R5 combined with a 31x31 local variance filter gave good results. Greenhill and Davies [15] employed 3x3 Laws' masks in conjunction with a neural network classifier. The output of the neural net is passed through a mode filter to remove small areas that have been incorrectly classified. They reported the results of a set of experiments on a Brodatz [2]

[2]Laws' texture energy measures are further explained in Sec. 2.3.

montage that used a variety of sizes of averaging and mode filters. They concluded that the optimum sizes for these two filters are 11x11 and 13x13 respectively, and that mode filters represent a valuable but under used technique. Harwood et al. [18] reported 92 and 94 percent success rates for classification of 120x120 samples of six Brodatz textures using L5E5 and L5S5 masks, respectively. du Buf et al. [3] used the variances of the masks' outputs within a relatively small 7x7 window and concluded that Laws' features were among the best tested out of a wide variety of texture measures.

In summary, in addition to Laws himself, a number of researchers have found these easily implementable texture energy measures to compare very well with alternative approaches.

1.3 Thesis Objective

The usual methods for image segmentation include edge detection, region growing, region splitting and merging, histogram thresholding, and clustering. Edge detection and histogram thresholding [36] methods work well with gray level images, but they may not be suitable for color images because color components should not be processed separately in view of their interdependence. Neighborhood-based methods, such as region growing, only use local information, while global methods, such as feature-space clustering, do not take advantage of local spatial knowledge [36]. Compared with these classical methods, the artificial neural network approach has the advantage of parallel processing (with appropriate hardware), robustness, noise tolerance, and adaptability. Self-organizing algorithms provide a framework for nonsupervised classification, the output of which can be used for controlled training of the next layer.

In this thesis, we propose a method for the segmentation of natural scenes by a two-stage hierarchical self-organizing map. The motivation to choose this architecture also arose from the need to develop a strategy for color image segmentation for which the amount of training data may be limited. The focus of this thesis is in the domain of photographic images with an essentially unlimited range of topics.

1.4 Thesis Organization

Chapter 2 gives a discussion on the strategies utilized in this approach. Chapter 3 gives the details of the proposed approach. Chapter 4 provides indicative results for the proposed approach and also provides a qualitative comparison with two other segmentation approaches. Chapter 5 discusses the conclusions and provides direction for future research.

CHAPTER 2

STRATEGIES

2.1 Color Models

Although the process followed by the human brain in perceiving and interpreting color is a physiopsychological phenomenon that is not yet fully understood, the physical nature of color can be expressed on a formal basis supported by experimental and theoretical results [14]. A color model (also called color space or color system) facilitates such an expression of colors in a standard, generally accepted way. In essence, a color model is a specification of a coordinate system and a subspace within that system wherein each color is represented by a single point [14].

2.1.1 RGB Color Model

Due to its absorption characteristics, the human eye perceives colors as variable combinations of the so-called primary colors red (R), green (G) and blue (B). This forms the basis for the RGB color model, which is shown in Figure 2.1. Each color appears in its primary spectral components of red, green and blue. This model is based on the Cartesian coordinate system.

Figure 2.1: The RGB color cube [14].

Figure 2.2 depicts the schematic for the RGB color cube. In this system, black is at the origin, while white is at the corner farthest from the origin. The gray scale extends

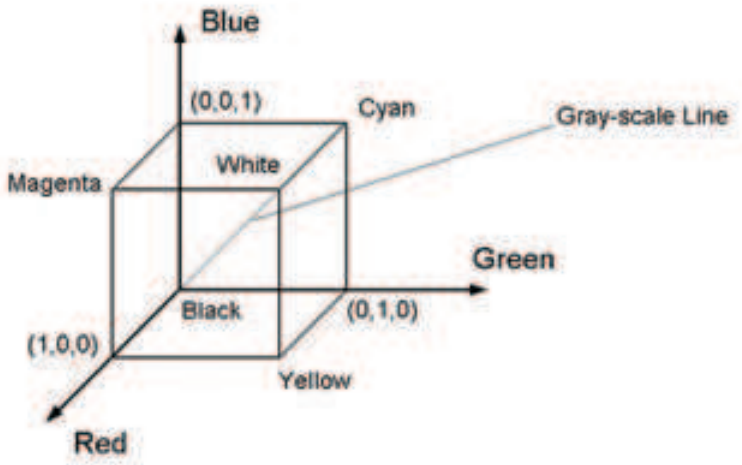

Figure 2.2: The schematic of the RGB color cube [14]

from black to white along the line joining these two vertices. The different colors inside this model are points on or inside the cube, and they are defined by vectors extending from the origin.

The RGB color model is a hardware-oriented model and is most commonly used for television systems and a broad selection of digital cameras. Video monitors display color images by modulating the intensity of the three primary colors (red, green and blue) at each pixel of the image [10, 39]. In spite of its popularity, the RGB model is not suitable for color image segmentation due to the high correlation among the R, G, and B components [32, 42]. Also, color difference is not represented on a uniform scale. Hence, it is impossible to evaluate the similarity of two colors based on their distance in the RGB color space.

2.1.2 CIE ($l^*a^*b^*$) Color Model

The Commission International de l'Eclairage (CIE) color system, shown in Figure 2.3 was developed to represent perceptual uniformity and thus meet the physiopsychological needs of a human observer.

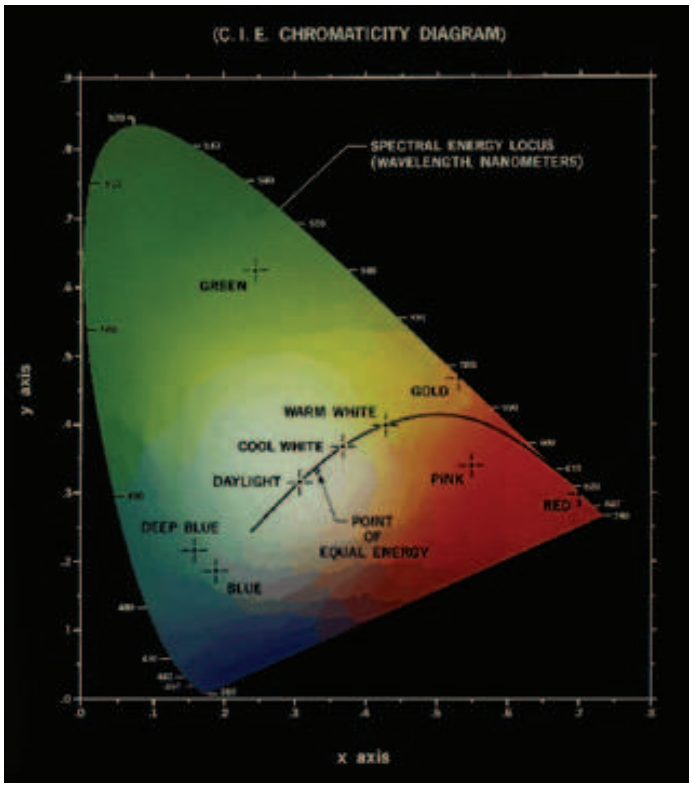

Figure 2.3: The CIE chromaticity diagram [14].

The tristimulus X, Y, Z form the primaries in this color model. This tristimulus can be obtained by a linear transformation from the corresponding RGB coordinates. The transformation matrix can be given as [6]:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \begin{pmatrix} 0.607 & 0.174 & 0.200 \\ 0.299 & 0.587 & 0.114 \\ 0.000 & 0.066 & 1.116 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

Once the X, Y, Z tristimulus coordinates are known, conversion to the CIE ($l^*a^*b^*$) color space can be achieved using the following nonlinear transformation:

$$\begin{aligned} l^* &= 116\sqrt[3]{\frac{Y}{Y_0}} - 16 \\ a^* &= 500[\sqrt[3]{\frac{X}{X_0}} - \sqrt[3]{\frac{Y}{Y_0}}] \\ b^* &= 500[\sqrt[3]{\frac{Y}{Y_0}} - \sqrt[3]{\frac{Z}{Z_0}}] \end{aligned}$$

where X_0, Y_0, Z_0 are the X,Y,Z values for standard white. Each color can, therefore, be regarded as a point in the three dimensional ($l^*a^*b^*$) color space. Color differences can be

calculated as:

$$\Delta E_{ab} = \sqrt{(\Delta l^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2} \tag{2.1}$$

The CIE ($l^*a^*b^*$) space can control color and intensity information more independently and simply than RGB primary colors. This model is especially efficient in the measurement of small color differences as it allows direct color comparison based on geometric separation within the color space.

2.2 Hierarchical Self-Organizing Maps

The hierarchical self-organizing map (HSOM) is a variant of the self-organizing map (SOM), proposed by Kohonen [26] and shown in Figure 2.4. The basic idea of an SOM is simple yet effective. An SOM consists of M neurons located on a regular low-dimensional grid, usualy 1- or 2-dimensional.

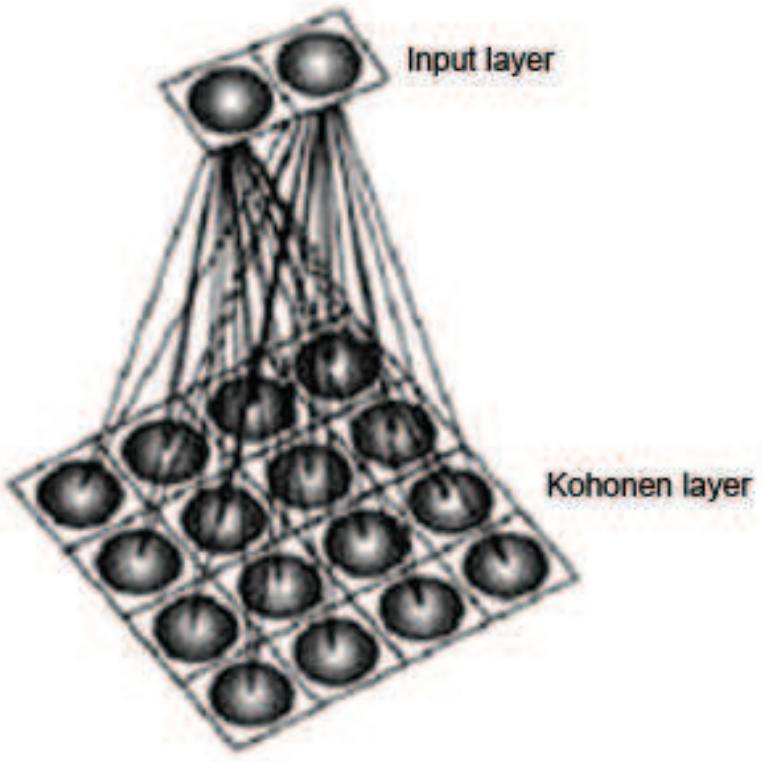

Figure 2.4: A conceptual view of the SOM: shows the connection between the input vectors and the neurons in the Kohonen map.

Higher dimensional grids are possible, but they are not generally used since their visualization is problematic. The lattice of the grid is either hexagonal or rectangular; see Figure 2.5.

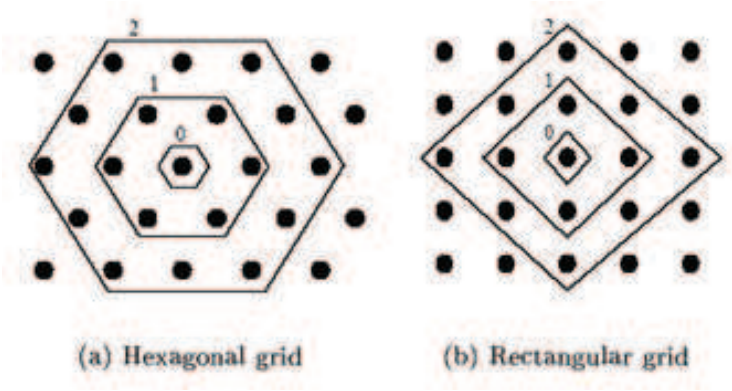

Figure 2.5: The SOM neighborhood sets (at radius 0, 1, 2) of the centermost unit for hexagonal and regular lattice.

The training mechanism for the basic SOM is iterative. Each neuron i has a d-dimensional prototype vector $\mathbf{m}_i = [m_{i1}, ..., m_{id}]$. At each training step, a sample data vector $\mathbf{x}$ is randomly chosen from the training set. Distances between $\mathbf{x}$ and all prototype vectors are computed. The best-matching unit (BMU), denoted here by b, is the map unit with prototype closest to $\mathbf{x}$:

$$\|\mathbf{x} - \mathbf{m}_b\| = \min_i \{\|\mathbf{x} - \mathbf{m}_i\|\} \tag{2.2}$$

Next, the prototype vectors are updated. The BMU and its topological neighbors are moved closer to the input vector in the input space, as shown in Figure 2.6. The update rule for the prototype vector of unit i is:

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha(t) h_{bi}(t)[\mathbf{x} - \mathbf{m}_i(t)], \tag{2.3}$$

where t denotes time, $\alpha(t)$ is learning rate and $h_{bi}(t)$ is a neighborhood kernel centered on the winner unit. An example of a kernel would be the Gaussian kernel:

$$h_{bi}(t) = e^{-\frac{\|\mathbf{r}_b - \mathbf{r}_i\|^2}{2\sigma^2(t)}}, \tag{2.4}$$

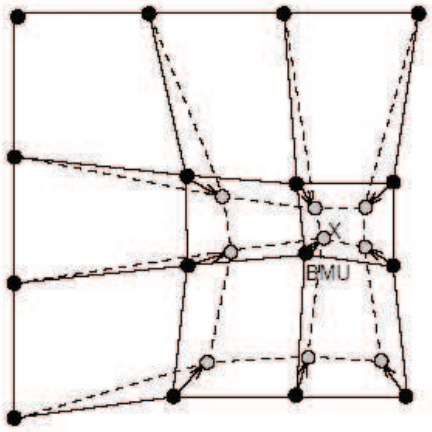

Figure 2.6: Updating the BMU and its neighbors towards the input sample marked with x: The neuron position before and after the update is represented by the black and the gray dots, respectively. The lines show neighborhood relations.

where $\mathbf{r}_b$ and $\mathbf{r}_i$ are positions of neurons b and i on the SOM grid, and $\sigma(t)$ is the neighborhood radius. Both the learning rate $\alpha(t)$ and neighborhood radius $\sigma(t)$ decrease monotonically with time.

During training, the SOM behaves like a flexible net that folds onto the "cloud" formed by the training data. Because of the neighborhood relations, neighboring prototypes are pulled in the same direction, and, thus, prototype vectors of neighboring units resemble each other. Thus, the goal of an SOM is to create a topologically (i.e. locally) ordered mapping of the input data. The primary benefits of an SOM are:

- *Robustness.* Assuming the neighborhood function reaches far enough, as in the case of a Gaussian neighborhood function, the SOM is very robust as all prototypes are affected by all data samples.
- *Local tuning.* The topological ordering works primarily in the neighborhood of each map unit, and, therefore, tunes locally to the data density.
- *Ease of visualization.* The regular map grid makes it easy to build efficient visualizations and their user interfaces.

The hierarchical SOM (HSOM) can be defined as a two-layer SOM whose operating principle is:

- For each input vector $\mathbf{x}$, the best matching unit is chosen from the first layer map, and its index b is input into the second layer;

- The best matching unit for input b is chosen from the second layer map, and its index is the output of the network.

In clustering terminology, while the first SOM layer forms one large cluster of all the data samples so that the total distance of the samples from the cluster is minimized, the second map in the HSOM splits the large cluster into parts of equal size. The advantage of using the HSOM is that each high dimensional data vector is mapped to a low-dimensional discrete value so that comparing the values implicitly contains comparison of the original distances.

2.3 Laws' Texture Energy Measures

Laws [28, 29] developed a set of two-dimensional masks derived from five simple one-dimensional filters. They are:

$$\begin{array}{lcccccccc} \text{L5} & = & (& 1 & 4 & 6 & 4 & 1 &) \\ \text{E5} & = & (& -1 & -2 & 0 & 2 & 1 &) \\ \text{S5} & = & (& -1 & 0 & 2 & 0 & -1 &) \\ \text{W5} & = & (& -1 & 2 & 0 & -2 & 1 &) \\ \text{R5} & = & (& 1 & 4 & 6 & -4 & 1 &) \end{array}$$

Note that the mnemonics stand for Level, Edge, Spot, Wave and Ripple. Laws convolved these masks with the transposes of each other to provide a set of symmetric and anti-symmetric center-weighted masks with all but the Level filters being zero-sum. Figure 2.7 depicts four of Laws' most successful masks. The above masks are convolved with the original image to produce a number of images that are themselves passed through a second stage which Laws termed a "macro statistic" [28]. This consists of a moving window estimation of the energy within the images. Thus Laws' feature measures estimate the energy within the passband of their associated filters. Consequently, he called his operators "texture energy measure." He noted that variance is defined in terms of a sum of squares partly for mathematical convenience and proposed as an alternative, a cheaper but

-1	-2	0	2	1
-4	-8	0	8	4
-6	-12	0	12	6
-4	-8	0	8	4
-1	-2	0	2	1

L5E5

-1	0	2	0	-1
-2	0	4	0	-2
0	0	0	0	0
2	0	-4	0	2
1	0	-2	0	1

E5S5

1	-4	6	-4	1
-4	16	-24	16	-4
6	-24	36	-24	6
-4	16	-24	16	-4
1	-4	6	-4	1

R5R5

-1	0	2	0	-1
-4	0	8	0	-4
-6	0	12	0	-6
-4	0	8	0	-4
-1	0	2	0	-1

L5S5

Figure 2.7: Four of Laws' most successful masks.

approximate measure: the average of the absolute values (ABSAVE). He found this to be just as successful, with less computation.

Figure 2.8 shows the grayscale version of an image used in this research.

Figure 2.8: The grayscale version of original image.

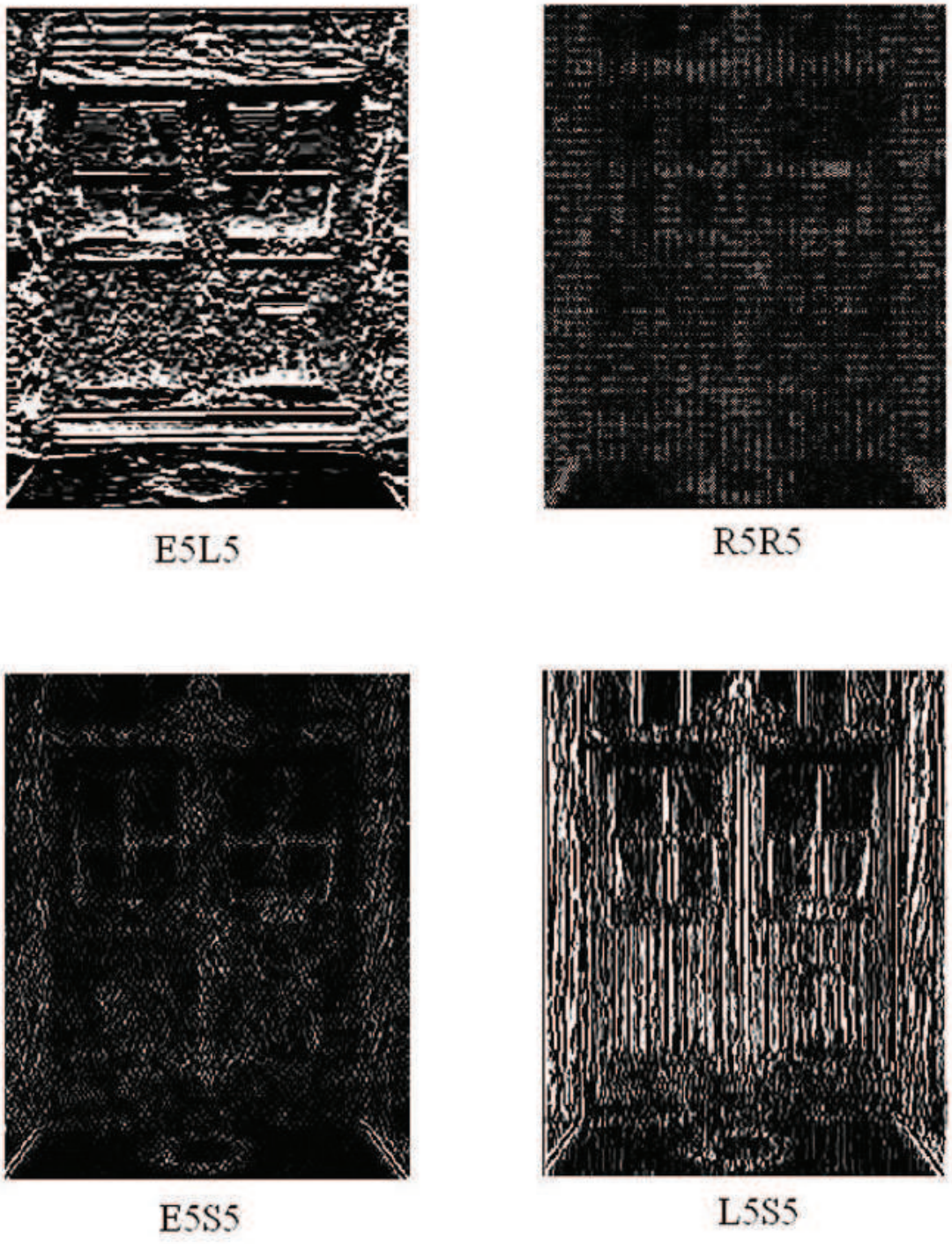

Figure 2.9: Filtered image planes: the result of applying Laws' four most successful masks.

Figure 2.9 shows the results of filtering each image with the four most important center-weighted masks mentioned above. As seen from the results, E5L5 is a horizontal edge mask. It enhances the horizontal structure in textures. R5R5 is a high-frequency spot detector. It produces a grainy feature plane that is very difficult to reproduce. E5S5 is a peculiar V-shaped mask that responds best to textures with a low correlation. E5S5 also seems to enhance the diagonal edges in the image. L5S5 is a vertical line detector. It enhances the vertical edges in the image, particularly the repetitive ones, such as the structural lines in the wooden door.

Figure 2.10 shows the effect of the ABSAVE texture energy transform and subsequent normalization with the standard deviation. Careful examination of the images shows that the different parts of the scene also have different relative brightness in the different texture energy planes.

Thus, Laws' texture energy transforms are a class of spatial-statistical transforms that effectively discriminate among texture fields. The essence of this approach is local measurement of the energy passed by a set of symmetric and anti-symmetric filters. Even though other descriptors for texture exist, there is no doubt that these measures can be used for segmentation of natural images.

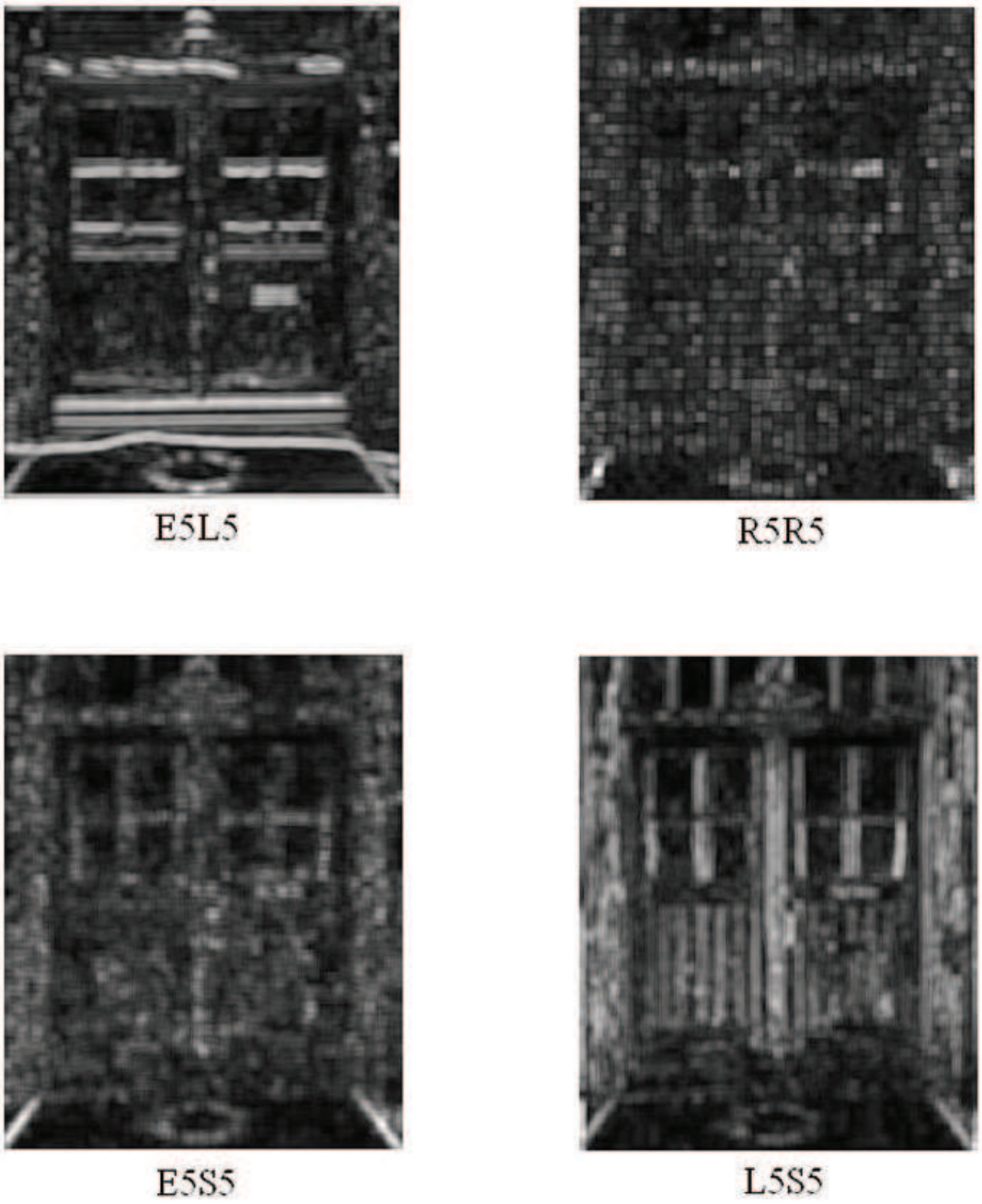

Figure 2.10: Texture energy planes: the result of normalization after applying the ABSAVE transform.

CHAPTER 3

THE PROPOSED ALGORITHM

3.1 Flowchart

Figure 3.1 shows the flowchart of the algorithm proposed in this research.

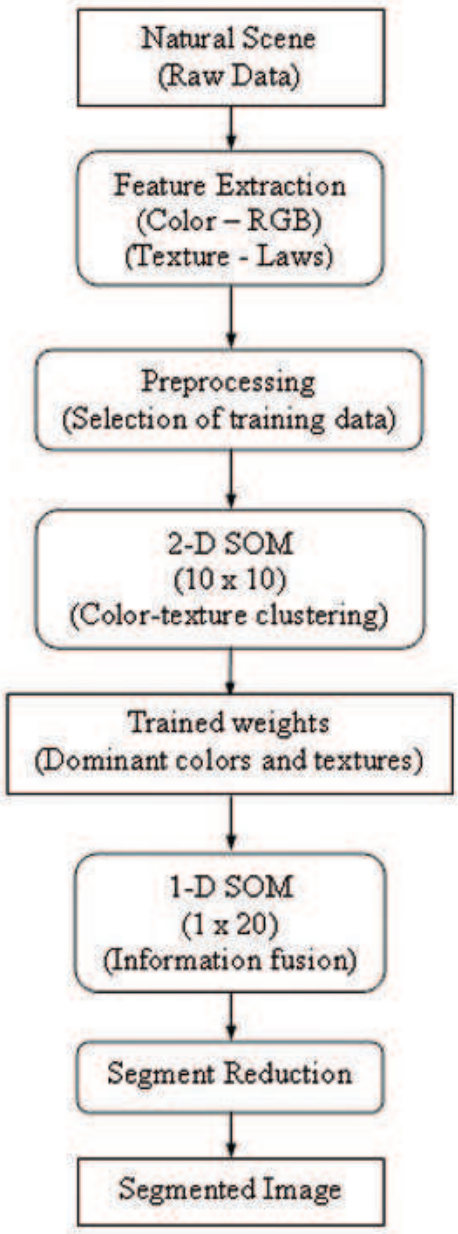

Figure 3.1: The flowchart of the proposed algorithm.

The image shown in Figure 3.2 is used as an example to demonstrate the results of the various stages of the algorithm.

Figure 3.2: Example image with an RGB description (151 x 172 x 194). The size of the image is 190 x 180 pixels.

3.2 Feature Extraction

The objective of this stage is to represent each pixel as a vector of features suitable for input into the SOM for training and subsequent classification. For this research, each pixel is represented as a seven-dimensional vector: $\{r, g, b, e5l5, e5s5, r5r5, l5s5\}$. At the end of this stage, the image is represented by a matrix in which each row is a feature and each column is a pixel location. Assuming that the image has M rows and N columns, the input data for the SOM is a matrix of feature values for all the pixels in the image, formulated as $7 \times (M \times N)$ matrix.

3.2.1 Color Features

The first three components of the feature vector are R, G, B values for every pixel in the image. Figure 3.3 shows the R, G, B color planes for the example image in Figure 3.2. Each of the color planes in the image is reshaped into a row-vector, and the three vectors are then vertically concatenated to form the first three rows of the input matrix.

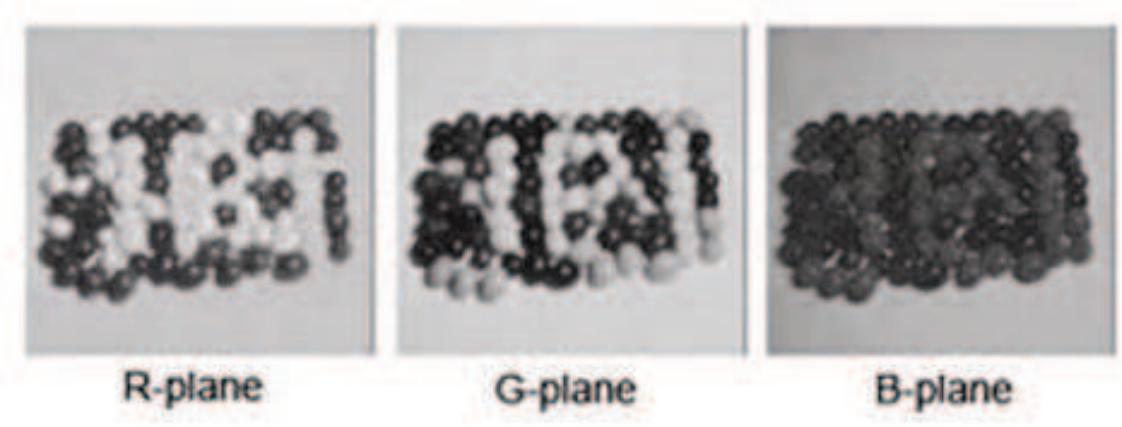

Figure 3.3: R, G, B planes for the example image.

3.2.2 Texture Features

The next four rows of the input matrix are obtained by applying the Laws' texture energy measures as explained in Section 2.3. The outputs of the four filters are reshaped into row vectors and concatenated with the input matrix. The results after applying Laws' masks and performing the ABSAVE operation on the example image are shown in Figure 3.4.

3.3 Preprocessing

The literature related to selection of data for training the SOM is scarce. Random selection is the most commonly used policy. While random selection ensures an unbiased collection of training samples, it does not always provide the optimal set of training samples. In the case of image segmentation, the pixels on and around the boundary of the perceptual segments provide the most information and should be used for training. As a step in this direction, a novel approach for selecting training samples is proposed in this thesis. This approach may not be optimal in its selection strategy, but it is certainly more intuitive than random selection methods. The selection criteria is based on a definition of homogeneity proposed in [9].

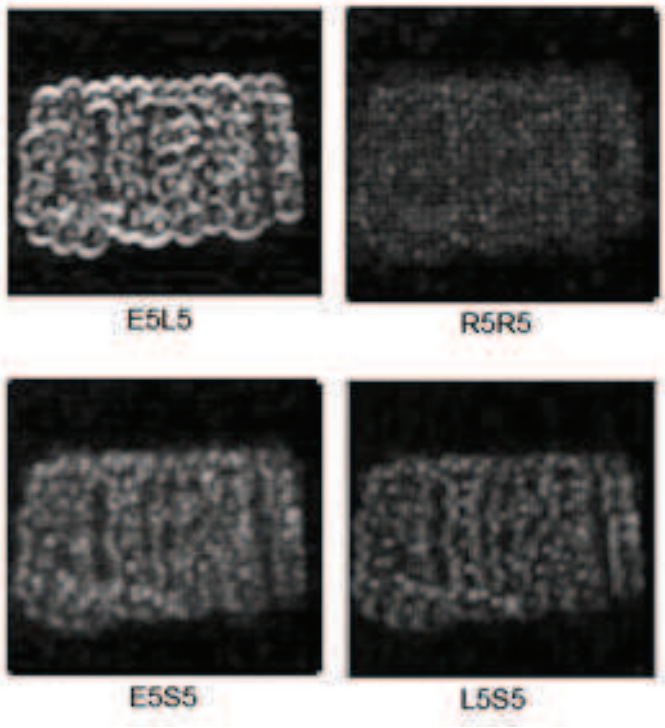

Figure 3.4: The result of applying four of Laws' most successful masks followed by the ABSAVE operation to the example image.

3.3.1 Homogeneity Measure

Homogeneity is mainly related to the local information of an image and reflects how uniform an image region is. [9] defines homogeneity as a composition of the following five components:

- *Edge_value.* Edge value measures abrupt changes in gray levels. At each pixel location P_{ij}, the magnitude of the gradient obtained by applying the Sobel edge detector is used as the measurement

$$e_{ij} = \sqrt{s_1^2 + s_2^2} \tag{3.1}$$

  where, s_1 and s_2 are results corresponding to the row and column masks, respectively.

- *Standard_deviation.* Standard deviation measures the contrast within the local region. It can be calculated as the difference between the gray level of a target pixel and the mean of the gray levels of the surrounding pixels in the region. The standard

deviation can be calculated as

$$v_{ij} = \sqrt{\frac{1}{d^2} \sum_{p=i-\frac{d-1}{2}}^{i+\frac{d-1}{2}} \sum_{q=j-\frac{d-1}{2}}^{j+\frac{d-1}{2}} (g_{pq} - \mu_{ij})^2} \tag{3.2}$$

where $i \geq 0$, $j \geq 0$, $p \geq M-1$, $q \geq N-1$ and μ_{ij} is the mean of the gray levels within the window w_{ij}. The value of μ_{ij} can be formulated as

$$\mu_{ij} = \frac{1}{d^2} \sum_{p=i-\frac{d-1}{2}}^{i+\frac{d-1}{2}} \sum_{q=j-\frac{d-1}{2}}^{j+\frac{d-1}{2}} g_{pq} \tag{3.3}$$

- *Entropy.* Entropy is a local image statistic that can be used to estimate the brightness variation in a region. The entropy of a pixel p_{ij} can be calculated as

$$h_{ij} = -\frac{1}{2 \log d} \sum_{k=1}^{L} P_k \log P_k \tag{3.4}$$

where P_k is the probability of the k^{th} gray level, which can be calculated as $\frac{n_k}{d^2}$; n_k is the total number of pixels with the k^{th} gray level; and L is the total number of gray levels in the window.

- *Skewness and Kurtosis.* Skewness is a measure of the symmetry of the distribution, and the Kurtosis is a measure of the impulsiveness of the distribution. The skewness can be calculated as

$$\gamma_{3ij} = \frac{\sum_{p=i-\frac{d-1}{2}}^{i+\frac{d-1}{2}} \sum_{q=j-\frac{d-1}{2}}^{j+\frac{d-1}{2}} (g_{pq} - \mu_{ij})^3}{(N-1)\sigma_{ij}^3} \tag{3.5}$$

The kurtosis can be formulated as

$$\gamma_{4ij} = \frac{\sum_{p=i-\frac{d-1}{2}}^{i+\frac{d-1}{2}} \sum_{q=j-\frac{d-1}{2}}^{j+\frac{d-1}{2}} (g_{pq} - \mu_{ij})^4}{(N-1)\sigma_{ij}^3} - 3 \tag{3.6}$$

where $N = d \times d$, d is the window size, and σ is the standard deviation over N observations. It can be calculated as

$$\sigma_{ij} = \sqrt{\frac{1}{N-1} \sum_{p=i-\frac{d-1}{2}}^{i+\frac{d-1}{2}} \sum_{q=j-\frac{d-1}{2}}^{j+\frac{d-1}{2}} (g_{pq} - \mu_{ij})^2} \tag{3.7}$$

where μ_{ij} is the mean of the gray levels within window w_{ij} and can be calculated with Equation 3.3

The *edge_value*, *standard_deviation*, *entropy*, *skewness* and *kurtosis* values are normalized to achieve computational consistency. After normalization, each component is in the range 0 to 1.

Homogeneity represents uniformness. The more uniform the local region surrounding a pixel is, the larger the homogeneity value for the pixel. Based on this concept, [9] provides the following definition of homogeneity:

$$\begin{aligned} HO(g_{ij}, w_{ij}) &= \overline{E(g_{ij}, w_{ij})} \times \overline{V(g_{ij}, w_{ij})} \\ &\quad \times \overline{H(g_{ij}, w_{ij})} \times \overline{R_3(g_{ij}, w_{ij})} \\ &\quad \times \overline{R_4(g_{ij}, w_{ij})} \\ &= (1 - E(g_{ij}, w_{ij})) \times (1 - V(g_{ij}, w_{ij})) \\ &\quad \times (1 - H(g_{ij}, w_{ij})) \times (1 - R_3(g_{ij}, w_{ij})) \\ &\quad \times (1 - R_4(g_{ij}, w_{ij})) \end{aligned} \tag{3.8}$$

where E, V, H, R_3, R_4 represents the normalized values, $0 \leq i \leq M-1$ and $0 \leq j \leq N-1$ and w_{ij} is the window centered at (i,j).

The homogeneity is also normalized to give β_{ij}. Finally, the nonhomogeneity is defined as

$$\begin{aligned} \psi_{ij} &= \psi(g_{ij}, w_{ij}) \\ &= 1 - \beta_{ij} \end{aligned} \tag{3.9}$$

3.3.2 Extension to RGB Images

The homogeneity measure defined above holds for grayscale images. This concept is extended to the domain of RGB images for use in the segmentation of natural scenes. Suppose $\beta_{R_{ij}}$, $\beta_{G_{ij}}$ and $\beta_{B_{ij}}$ are the homogeneity measures calculated for the pixel location

(i,j) in the R, G and B color planes respectively, the homogeneity measure for the pixel location (i,j) in the RGB domain can be defined as

$$\beta_{RGB_{ij}} = \sqrt{\beta_{R_{ij}}^2 + \beta_{G_{ij}}^2 + \beta_{B_{ij}}^2} \tag{3.10}$$

The nonhomogeneity measure in the RGB domain can then be calculated as

$$\psi_{RGB_{ij}} = 1 - \beta_{RGB_{ij}} \tag{3.11}$$

3.3.3 Selection of Training Samples

This nonhomogeneity measure can then be used to obtain training data by using the following steps:

1. The average nonhomogeneity value is calculated for the entire RGB image as

$$\mu_{\psi_{image}} = \frac{1}{MN} \sum_{p=0}^{M-1} \sum_{q=0}^{N-1} \psi_{RGB_{pq}} \tag{3.12}$$

2. The image is divided in blocks of arbitrary size d (in this research, $d = 15$), and the local average nonhomogeneity value for each block is calculated as

$$\mu_{\psi_{block}} = \frac{1}{d^2} \sum_{p=i-\frac{d-1}{2}}^{i+\frac{d-1}{2}} \sum_{q=j-\frac{d-1}{2}}^{j+\frac{d-1}{2}} \psi_{RGB_{pq}} \tag{3.13}$$

3. For every $d \times d$ block of the image, the number of pixels to be chosen for training is decided by the threshold

$$n_{training} = (\mu_{\psi_{image}} - \mu_{\psi_{block}}) * d^2$$

4. $n_{training}$ pixels locations are then randomly selected from the block to be part of the training data.

5. This process is repeated for all the blocks in the image to get pixel locations of all the training samples.

6. The vectors corresponding to these locations are then extracted from the SOM input matrix to form the final training set.

Selection of pixels based on the nonhomogeneity criterion ensures that the training dataset contains more pixels representing the diverse regions in the image, rather than those representing the homogeneous regions.

3.4 HSOM Training and Testing

In a conventional self-organizing map (SOM) network, the number of regions in the final segmented image depends on the number of neural units in the Kohonen layer, but it is very improbable that the number of regions in an image is known a priori. This significant shortcoming is overcome by implementing a hierarchical two-stage self-organizing network as a pattern classifier to group the output neurons into subsets, each of which corresponds to a discrete region. The number of clusters formed can thus be controlled, and this, in turn, limits the number of colors that will appear in the segmented image. The first stage provides a preliminary clustering of the input color image data in an unsupervised mode, thereby obviating the need for a large number of training samples that may not be available in practice. The second stage network, which is much smaller in size, is subjected to training on the basis that only prominent features are selected using global information, and the segmentation is performed according to these features. The model implemented in this thesis consists of two stages, each of which is an SOM network.

3.4.1 Geometry

Among the few geometrical structures that could be used to construct the two-dimensional array for the first stage, the square and the hexagonal are the practical possibilities. In a square-array structure, each neuron has eight neighbors and the distance to the diagonal elements is mathematically different from the distance to the other neighbors, leading to a lack of uniformity in the computation of the influence of the neighborhood on the neuron. In contrast, in the hexagonal array, each neuron has six equidistant neighbors, making the array mathematically attractive. It is also known that the regular hexagonal sampling of images produces optimum data for a given information content. Hence, the two-dimensional stage of the HSOM is implemented as a hexagonal array as shown in Figure 3.5. There are 100 output neurons arranged in a 10 x 10 grid to form a two-dimensional

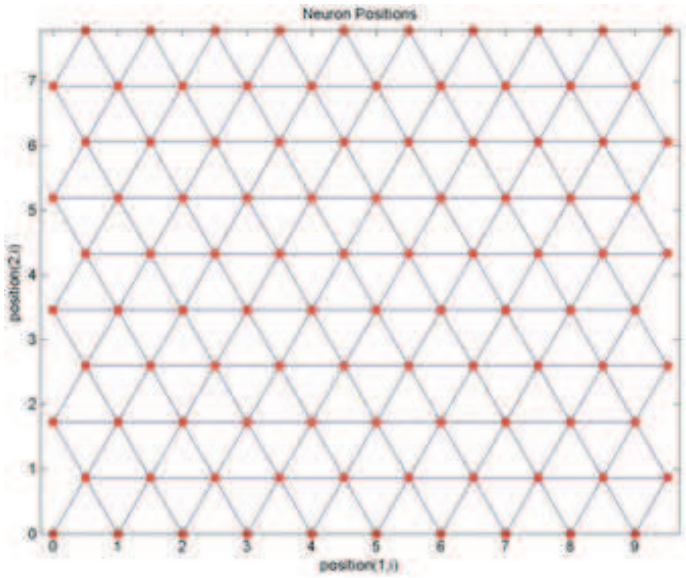

Figure 3.5: The hexagonal topology of the 2D SOM.

topological feature map. The choice of this size is a compromise that makes it possible for some distinctive features occurring infrequently to be sampled in the initialization phase and also allows for ease of convergence. The second stage is one-dimensional and contains 20 neurons, which implies that the 100 outputs of the first stage are now reduced to 20.

3.4.2 Stage 1: A Two-dimensional SOM

Both stages of the network share the same training principle mentioned in Section 2.2. Essentially, the neuron weights are initialized before training, the winning neuron and its neighbors within a calculated radius are updated via a learning rule, and learning continues until a convergence criterion is satisfied.

$Matlab^{©}$ initializes the weights for the Kohonen layer to the midpoint of the inputs. Thus, initially, all weights are assigned the value 0.5 (since the input ranges from 0 to 1). Figure 3.6 shows the initial weights of the color features for the two-dimensional SOM.

After undergoing training based on the strategy mention in Section 2.2, the neurons are clustered to represent the input data. The clustering of the color features is shown in Figure 3.7. The output at the end of the unsupervised learning for the first stage of the network is a coarsely segmented image, as shown in Figure 3.8 The weights of the neurons

Figure 3.6: The initial weights for the color features of the 2D SOM.

Figure 3.7: Color Clustering: The color-weights of the 2D SOM for the example image, after training has converged.

in this stage of the network tend to approximate the density function of the vector inputs obtained from the image in an orderly manner. However, the number of segments in the output of this stage is still quite large, i.e. 100 for the 10 x 10 network.

Figure 3.8: The segmentation result after applying the 2D SOM to the example image.

3.4.3 Stage 2: The One-dimensional SOM

The output of the first stage is fed to a one-dimensional array of neurons (with weights initialized to the midpoint) for training, and, as a consequence, the dominant features in the two-dimensional map are represented by neurons in the one-dimensional array. Thus, the most representative features are extracted from the original image, resulting in its segmentation. The output of this stage is a more cohesively segmented image, as shown in Figure 3.9. The output at the end of this stage further undergoes region merging to result in the final segmentation.

3.5 Region Merging

The output of the HSOM is, more often than not, an over-segmented image. Hence, the region-merging phase is implemented to combine regions that are similar to each other. As mentioned in Section 2.1.1, the RGB color model is not suitable for measuring color difference. Hence, the image is converted into the CIE ($l^*a^*b^*$) color space, and the region merging process proposed by [8] is carried out.

Each of the k ($k \leq 20$) regions generated by the HSOM phase represents a color. The color difference is calculated for all pairs of these colors according to Equation 2.1. This

Figure 3.9: The segmentation result after applying the 1D SOM to the example image.

results in a list of $\frac{k(k-1)}{2}$ color differences. The standard deviation of this list is calculated as

$$\sigma_v = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(D_i - \mu_d)^2} \tag{3.14}$$

where $n = \frac{k(k-1)}{2}$ is the number of color differences, μ_d is the mean of the differences and D_i is the i^{th} color difference in the list. The threshold for the color merging is set as

$$T_d = \mu_d - \sigma_v \tag{3.15}$$

In the region-merging algorithm, two regions with the smallest color difference are located. If this color difference is smaller than T_d, the two regions are merged. The color of the new region is calculated as the mean of the colors of its component regions. This operation is repeated until no color difference is smaller than the threshold T_d.

This phase generates the final segmentation result. The segmented image thus generated is seen to have more homogeneity within the regions and more disparity between them. The result of the region-merging phase on the example image is shown in Figure 3.10.

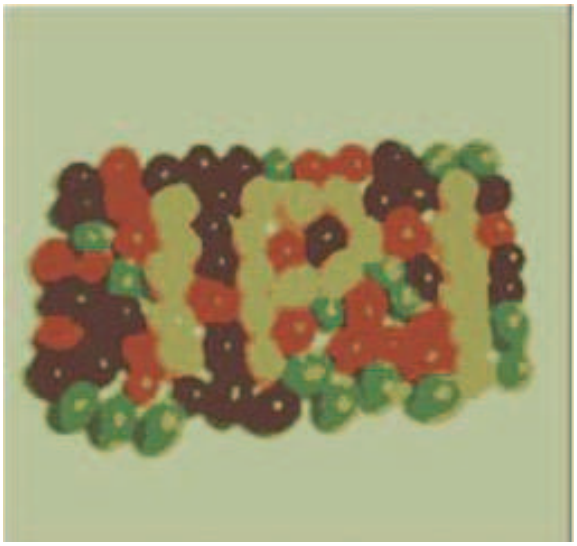

Figure 3.10: Final result after region-merging: segmented image with five colors.

CHAPTER 4

EXPERIMENTS AND DISCUSSIONS

4.1 Indicative Results

The proposed approach has been tested on a variety of color images including those of natural scenes. Segmentation results, examples of which are shown in Figures. 4.1 and 4.2, confirm that this approach is suitable for a wide range of images.

The original Landsat image in Figure 4.1(a) was acquired from the WWW, while the "stadium" image in Figure 4.1(d) is a part of a personal photo collection. Figures. 4.1(b) and 4.1(d) show the clustering of the two-dimensional SOM after training is complete for the Landsat and "stadium" images respectively. For ease of visualization, only color clustering is shown. Subsequently, Figures. 4.1(c) and 4.1(f) show the final results for the Landsat image (seven colors) and the "stadium" image (seven colors), respectively.

Similarly, the original "fireworks" image in Figure 4.2(a) is selected from a personal collection of photographs, while the "vegetables" image in Figure 4.2(d) was acquired from the WWW. The use of the SOM for segmentation seems more intuitive from the evident color clustering in Figures. 4.2(b) and 4.2(e). The final results in Figures. 4.2(c) and 4.2(f) show that the proposed approach is indeed viable and can be used for image segmentation of a wide variety of images.

4.2 Comparative Results

In order to demonstrate the performance of the proposed approach, we compare our results with two other approaches. [8] studies a novel hierarchical approach to color segmentation. In the two-phase procedure, the first phase results in the identification of uniform regions via multi-level thresholding on the homogeneity histogram. In the second phase, hue values for each uniform region are redefined to remove singularities. Finally, [8] employs a region merging-process to avoid over-segmentation. [7] proposes a fuzzy homogeneity and scale space approach for color image segmentation. [7] first fuzzifies the color image and then employs the fuzzy homogeneity histogram to process both local

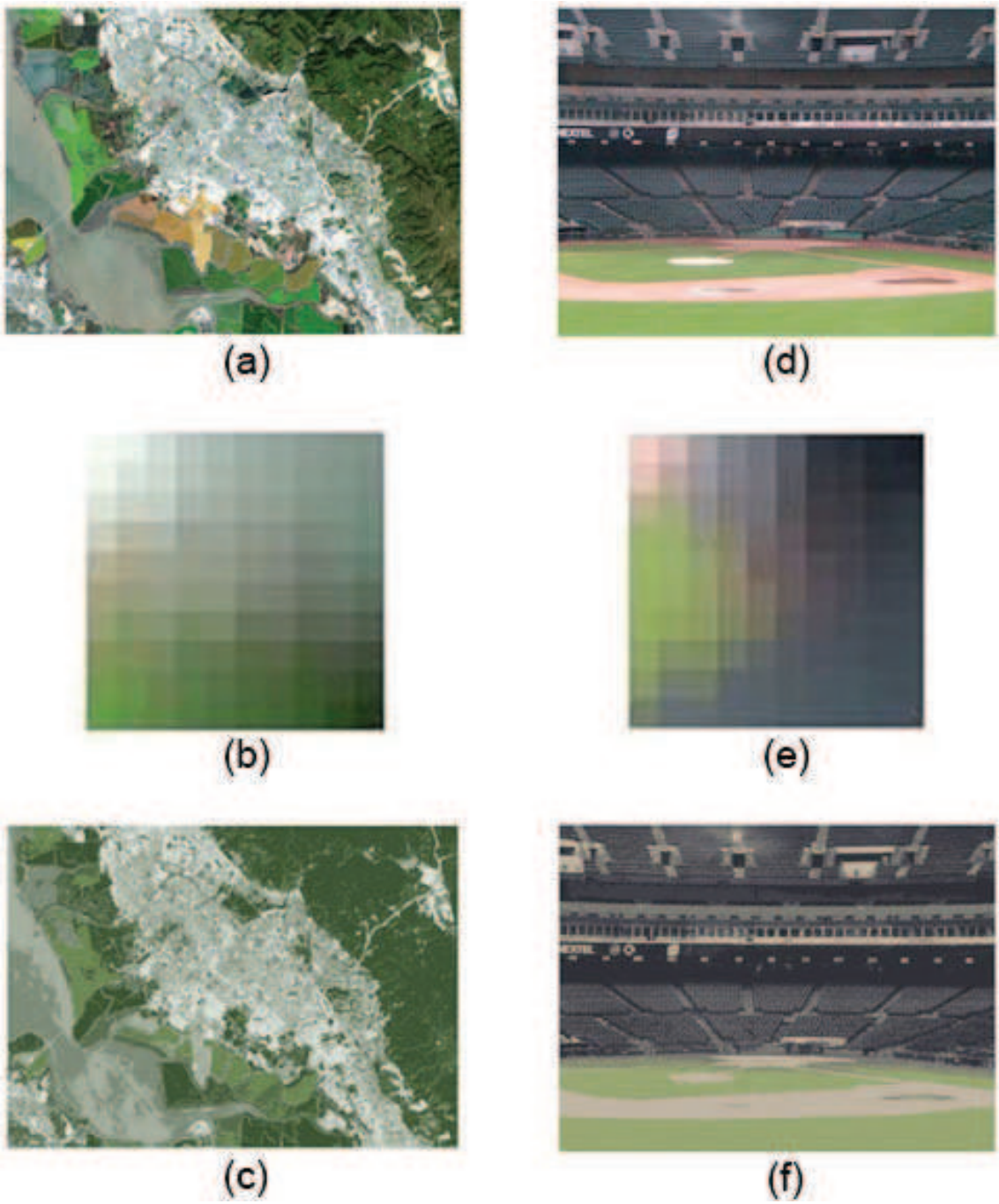

Figure 4.1: (a) The original Landsat image with size 650 x 467. (b) Color clustering of the 2D SOM for the Landsat image. (c) The segmented Landsat image with seven colors. (d) The original "stadium" image with size 480 x 360. (e) Color clustering of the 2D SOM for the stadium image. (f) The segmented stadium image with seven colors.

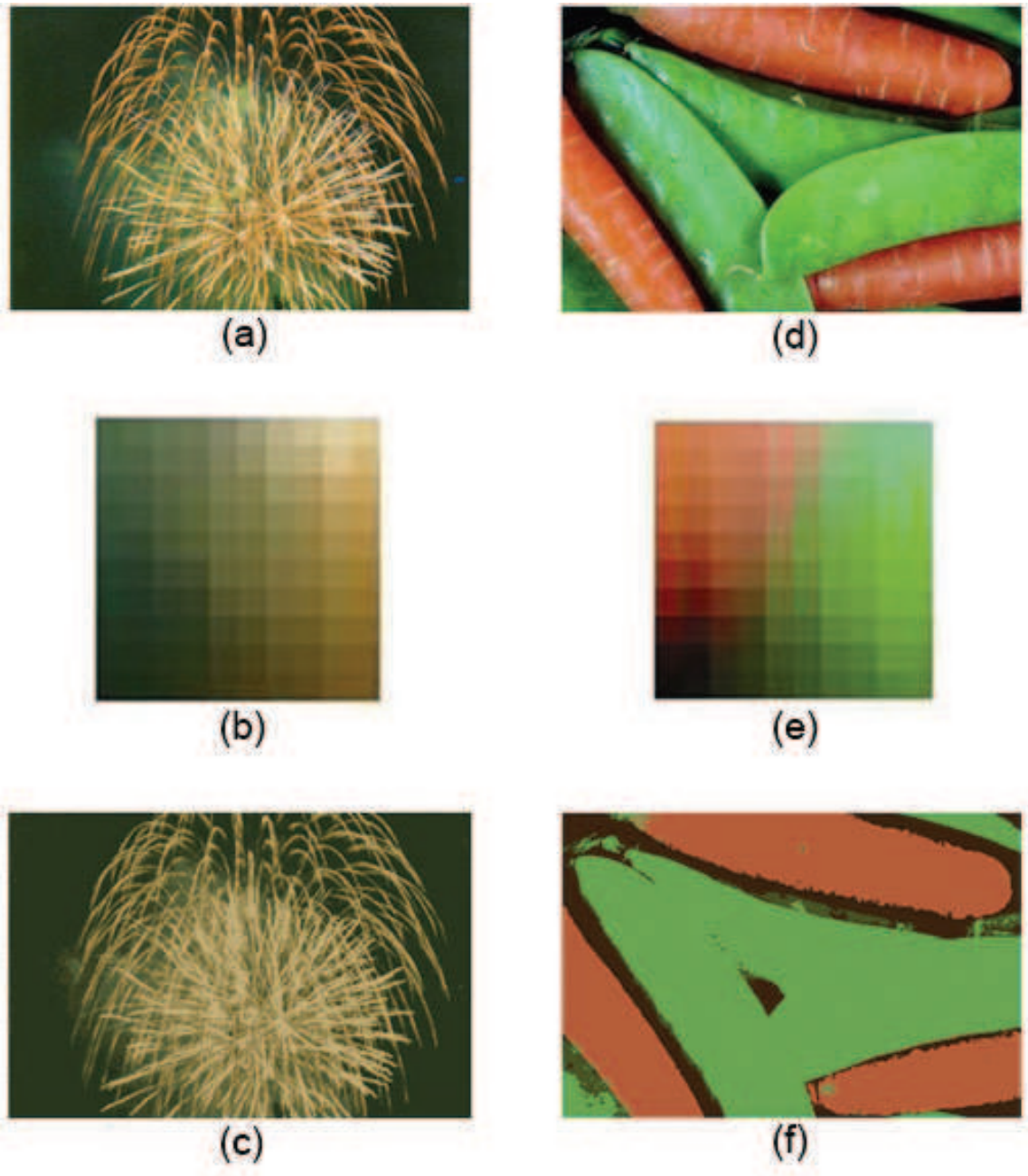

Figure 4.2: (a) The original "fireworks" image with size 594 x 395. (b) Color clustering of the 2D SOM for the fireworks image. (c) Segmented "fireworks" image with seven colors. (d) The original "vegetables" image with size 384 x 256. (e) Color clustering of the 2D SOM for the "vegetables" image. (f) The segmented vegetables image with five colors.

and global information. The scale-space filter (SSF) is then utilized to find the appropriate segments of the homogeneity histogram. Finally, [7] implements a fuzzy region-merging

process to restrict cluster sizes and avoid over-segmentation. The approach proposed in this thesis implements a region-merging process similar to the one implemented in [8] and [7]. Therefore, any improvement in the results can solely be attributed to the corresponding segmentation scheme.

Figures. 4.3 to 4.7 show the comparison between the segmentation performed by [8] and the proposed approach.

Figure 4.3(a) shows the original "kayak" image. The segmentation in Figure 4.3(b) is due to [8] and results in eight colors. Even though the segmentation is fairly homogeneous, some singularities still exist within the segments. For example, a close examination of the hull of the kayak reveals tessellation in the homogeneous region. This effect is eliminated in the result of the proposed approach, shown in Figure 4.3(c), with six colors. Similarly, the water region is more uniformly segmented by the proposed approach. Singularities, albeit pixel-level, are still visible in Figure 4.3(b).

Figure 4.4(a) shows the original "door" image. Figure 4.4(b) shows the results of segmentation using [8] (eight colors). The result of the proposed approach, shown in Figure 4.4(c), has six colors. Thus, the proposed approach is able to provide more homogeneous segments. This can be noted from the fact that the ground in front of the door results in one homogeneous segment due to the proposed method. The result of [8], for the same region, has singularities due to the overhead light object.

For the natural scene shown in the "mountain" image in Figure 4.5, [8] results in seven colors (Figure 4.5(b)), and the proposed approach yields nine colors (Figure 4.5(c)). The result of the proposed approach detects the presence of the cloud cover in the sky, and it preserves its symmetry in the reflection. In Figure 4.5(b), even though the clouds are seen in the reflection, they are not present in the sky. This asymmetry may not be considered a major discrepancy in the field of segmentation, but in an image retrieval system, this may result in the generation of inadequate annotations for the image.

The "sail" image shown in Figure 4.6(a) is a typical image. The shades in the evening sky are some of the most difficult to segment. The proposed method (five colors) captures

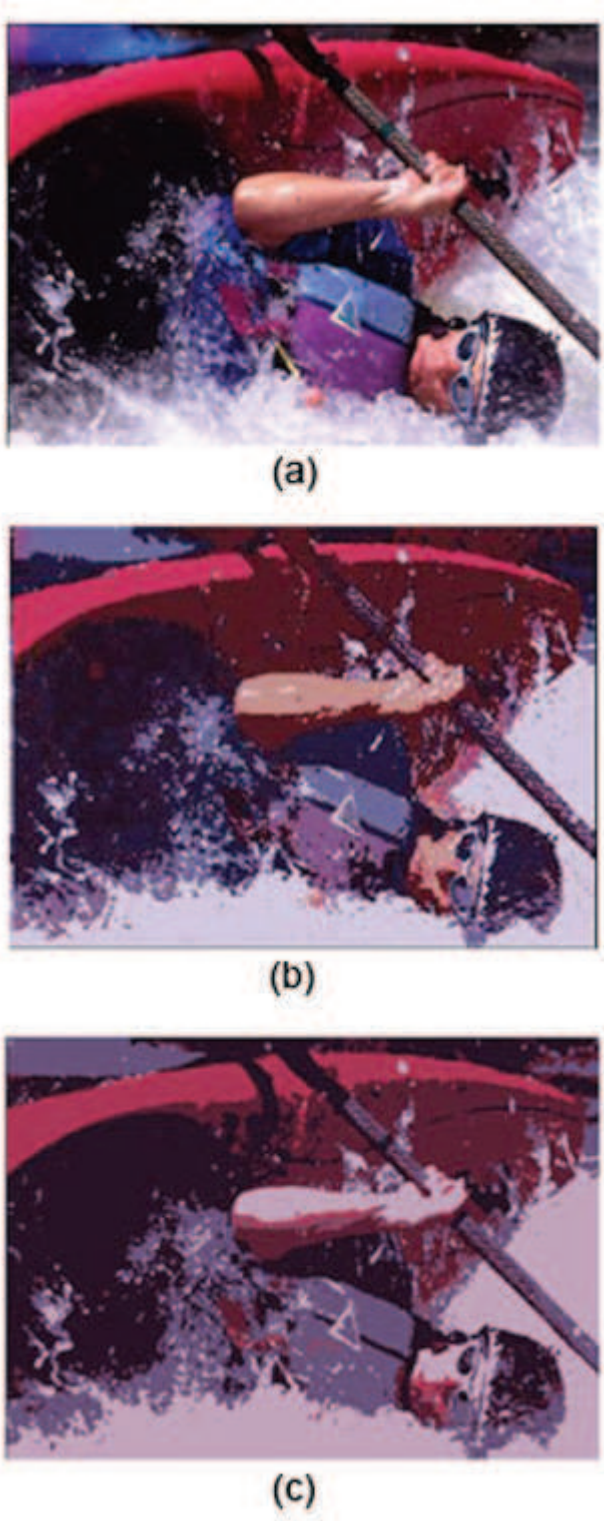

Figure 4.3: (a) Original "kayak" image. (b) Segmentation based on homogeneity and hue (8 colors). (c) Segmentation using the proposed approach (6 colors).

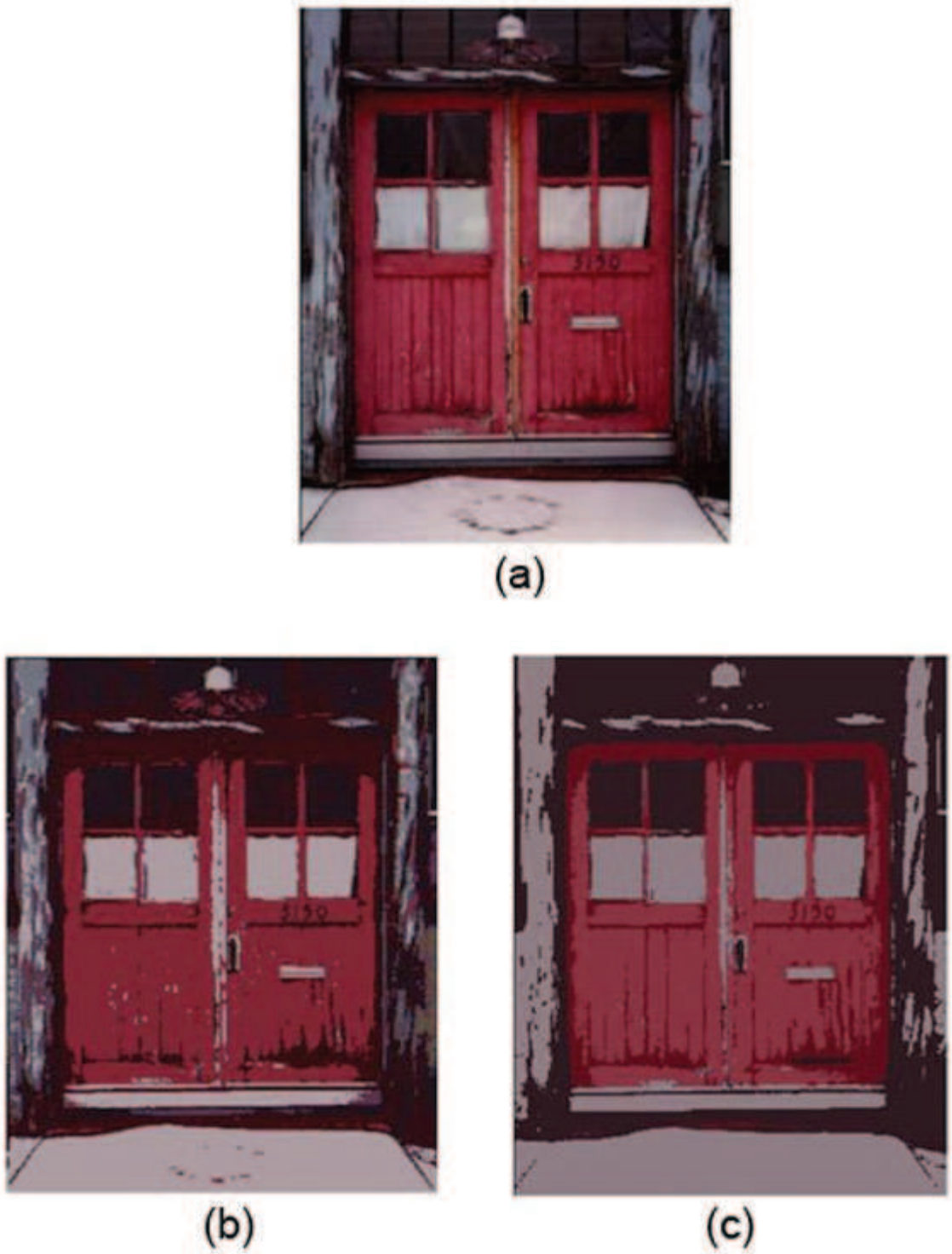

Figure 4.4: (a) The original "door" image. (b) Segmentation based on homogeneity and hue (eight colors). (c) Segmentation using the proposed approach (six colors).

the shades better than [8] (seven colors). Also, the effect of illumination on the water results in some lack of homogeneity in Figure 4.6(b). On the contrary, the proposed method is more robust and results in a more homogeneous segmentation, as shown in Figure 4.6(c).

An intuitive look at the "flowers" image in Figure 4.7(a) reveals that all the flowers are of the same type but in different stages of bloom. If segmentation is being used as part of a classification scheme, it is evident that the use of Figure 4.7(c) will result in a more accurate classification result as compared to Figure 4.7(b). Thus, seen in a broad scheme of things, the proposed approach will perform better than [8].

Figures. 4.8 to 4.12 compare the results of the approach mentioned in [7] with the results of the proposed approach.

The ripples in the water in the "duck" image in Figure 4.8(a) result in over-segmentation due to [7], as shown in Figure 4.8(b). In contrast, the result of the proposed approach, shown in Figure 4.8(c) is more intuitive and results in a more homogeneous segmentation with five colors.

Again, the shades in the mountain region in the "fire" image (Figure 4.9(a)), result in over segmentation due to [7]. As seen in Figure 4.9(b), the mountain region is not contiguous. The proposed approach is seen to be more intuitive and results in a homogeneous segmentation, as shown in Figure 4.9(c). Also, the contour of the smoke is approximated better by the proposed approach, as compared to [7].

A comparison of the segmentation results for the "peaches" image in Figure 4.10(a) reveals that the proposed approach results in smoother region boundaries, and, thus, has fewer misclassified boundary-pixels. This is evident in the absence of the brighter pixels at the boundary of the peaches (Figure 4.10(b)) in the result of the proposed approach (Figure 4.10(c)). The green tinge at the top-left and bottom-right corners of the original image is actually part of the background, and it should be classified thus. Segmentation due to [7] tends to retain this region, as shown in Fig. 4.10(b). The proposed approach results in a more homogeneous segmentation as shown in Figure 4.10(c).

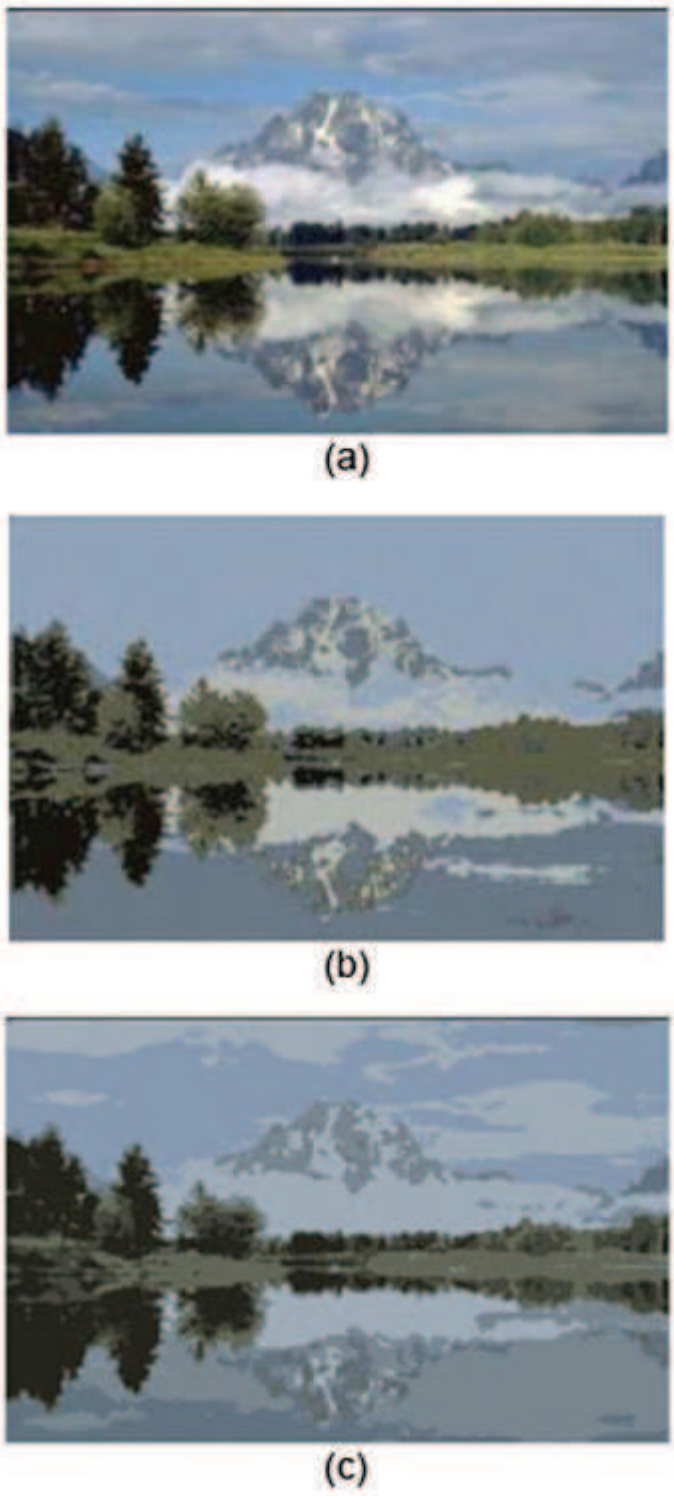

Figure 4.5: (a) The original "mountain" image. (b) Segmentation based on homogeneity and hue (seven colors). (c) Segmentation using the proposed approach (nine colors).

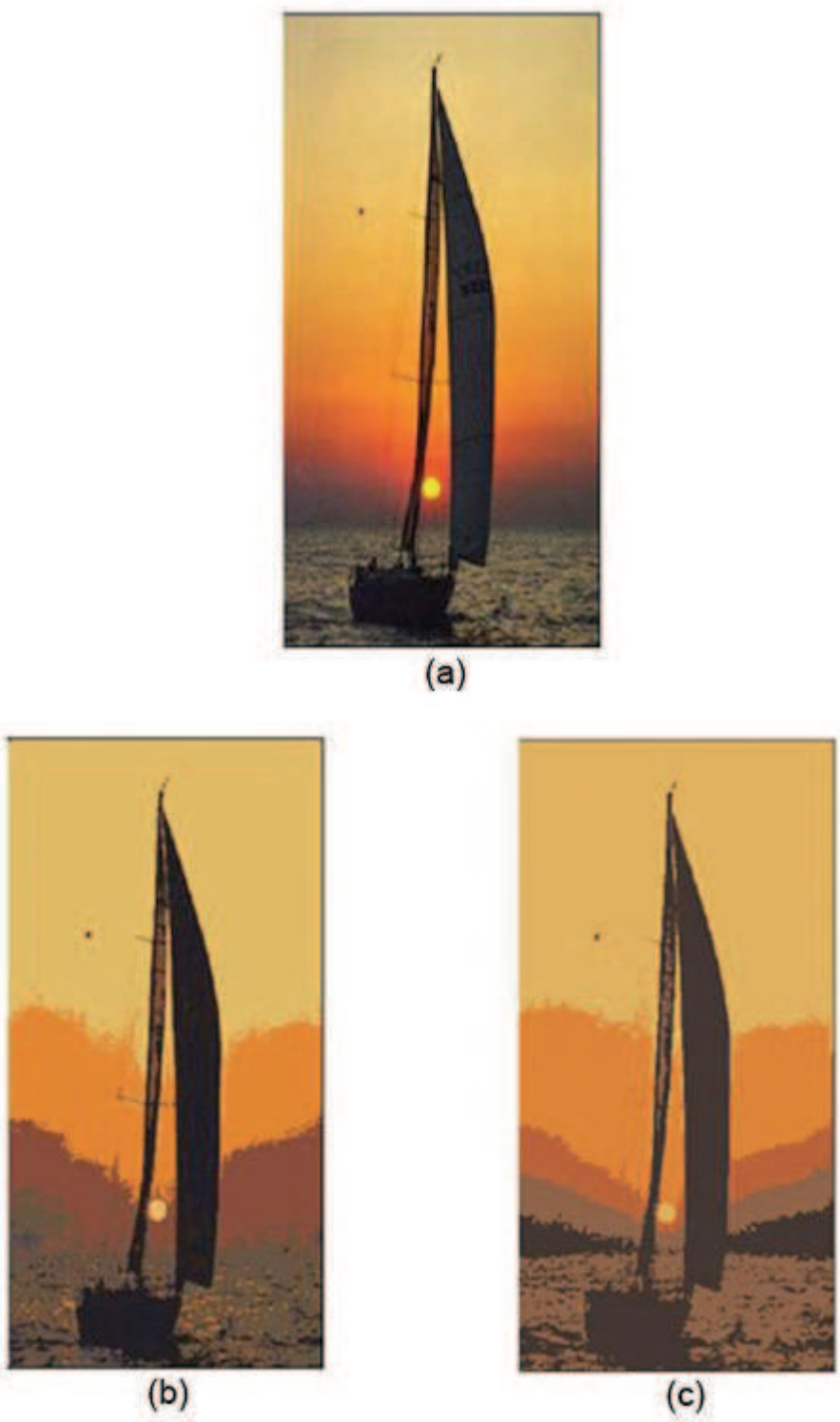

Figure 4.6: (a) The original "sail" image. (b) Segmentation based on homogeneity and hue (seven colors). (c) Segmentation using the proposed approach (five colors).

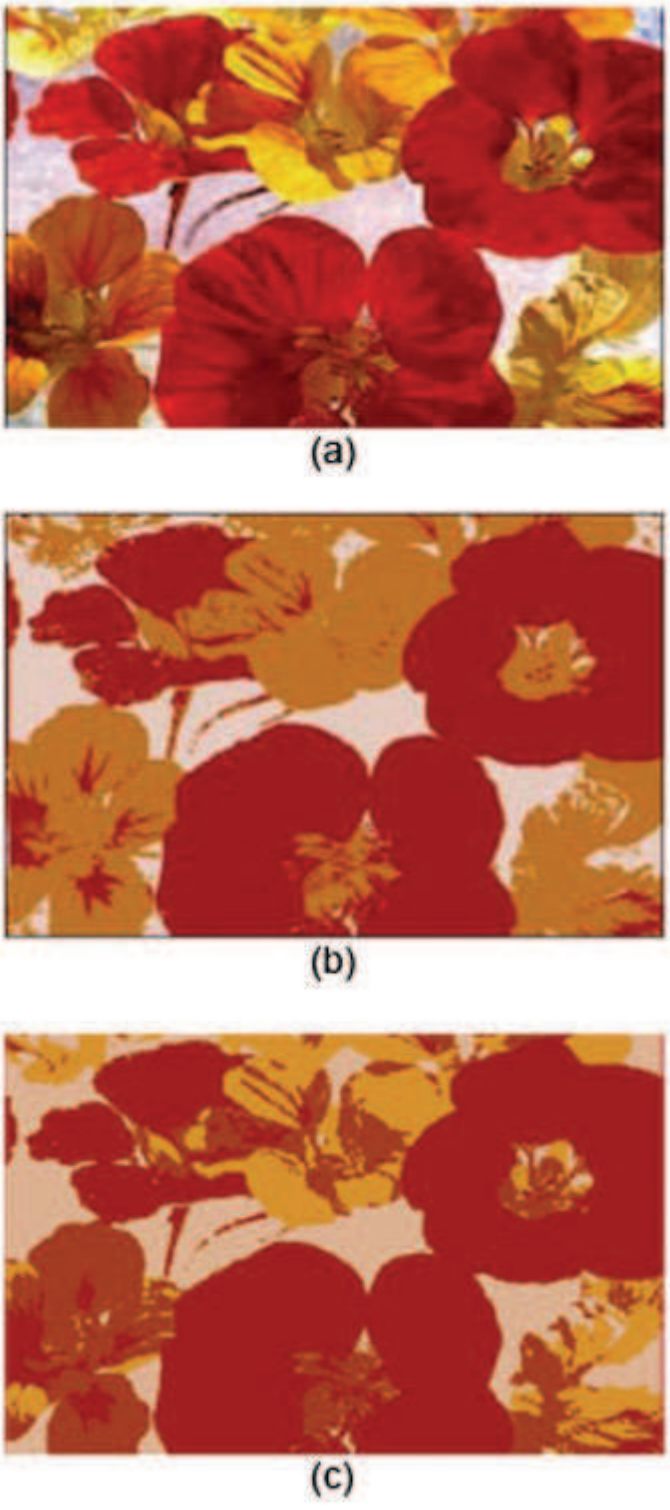

Figure 4.7: (a) The original "flowers" image. (b) Segmentation based on homogeneity and hue (three colors). (c) Segmentation using the proposed approach (five colors).

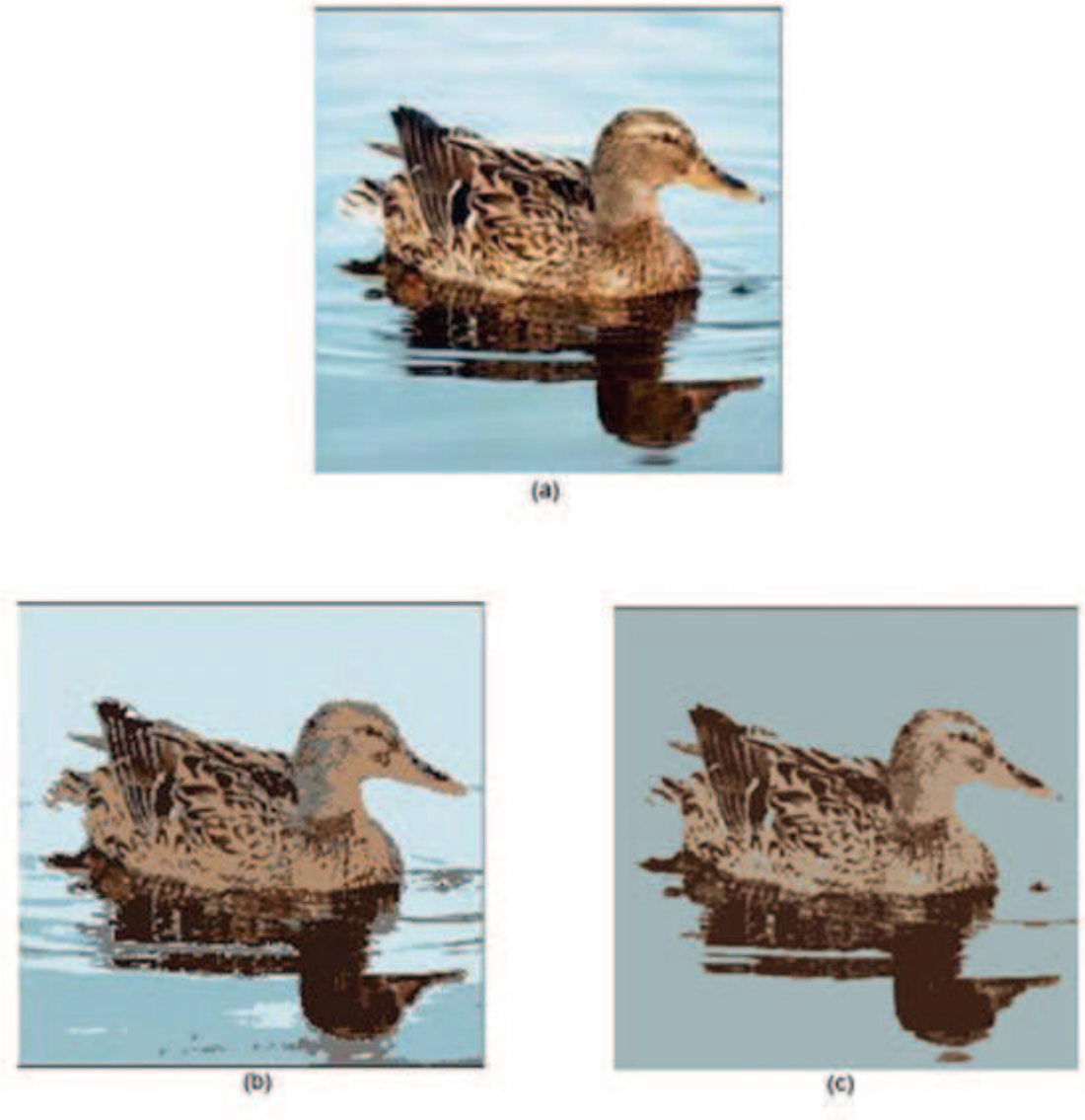

Figure 4.8: (a) The original "duck" image. (b) Segmentation result of the fuzzy homogeneity and SSF approach (seven colors). (c) Segmentation using the proposed approach (five colors).

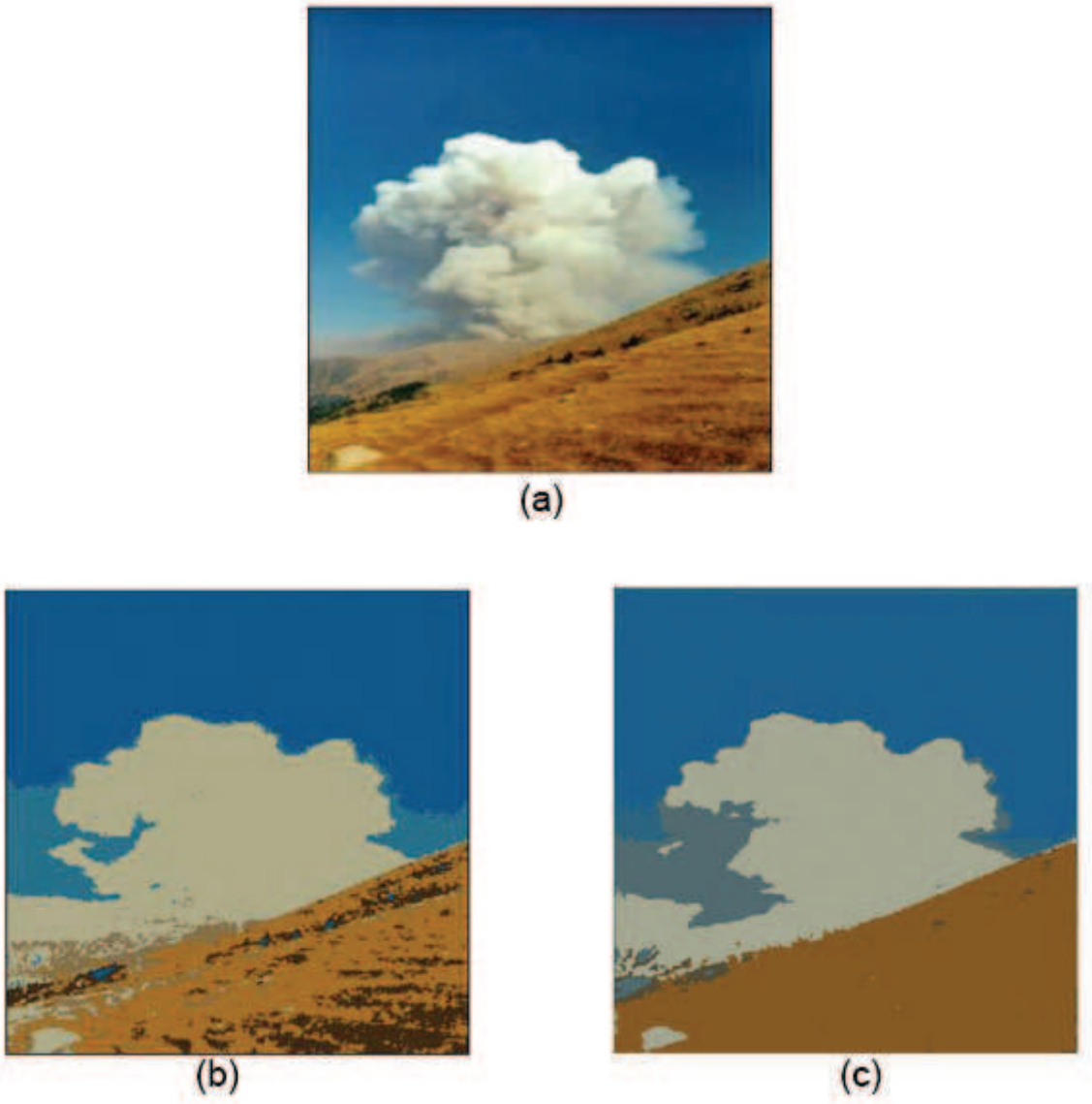

Figure 4.9: (a) The original "fire" image. (b) Segmentation result of the fuzzy homogeneity and SSF approach (five colors). (c) Segmentation using the proposed approach (six colors).

(a)

(b)

(c)

Figure 4.10: (a) The original "peaches" image. (b) Segmentation result of the fuzzy homogeneity and SSF approach (eight colors). (c) Segmentation using the proposed approach (five colors).

The "rose" image in Figure 4.11(a) has vivid colors and is difficult to segment. [7] approximates it to six colors (Figure 4.11(b)), but [7] is still not able to provide the homogeneity evident in the segmentation resulting from the proposed approach (Figure 4.11(c)). As mentioned earlier, the result of the proposed approach will be more valuable for the purpose of classification.

Finally, for the "salad" image in Figure 4.12(a), segmentation due to the proposed approach is more homogeneous. The boundaries between the regions (notice the effect of the shadow of the plate on the table) are more clearly defined in Figure 4.10(c) than in Figure 4.10(b).

The above comparison leads to the belief that the proposed approach performs the task of segmentation more intuitively than the ones it is compared with. This fact has important implications, for segmentation is often used as an elementary block in many applications, such as image retrieval and image annotation. The intuitive results of the proposed approach will prove to be much more beneficial in such systems.

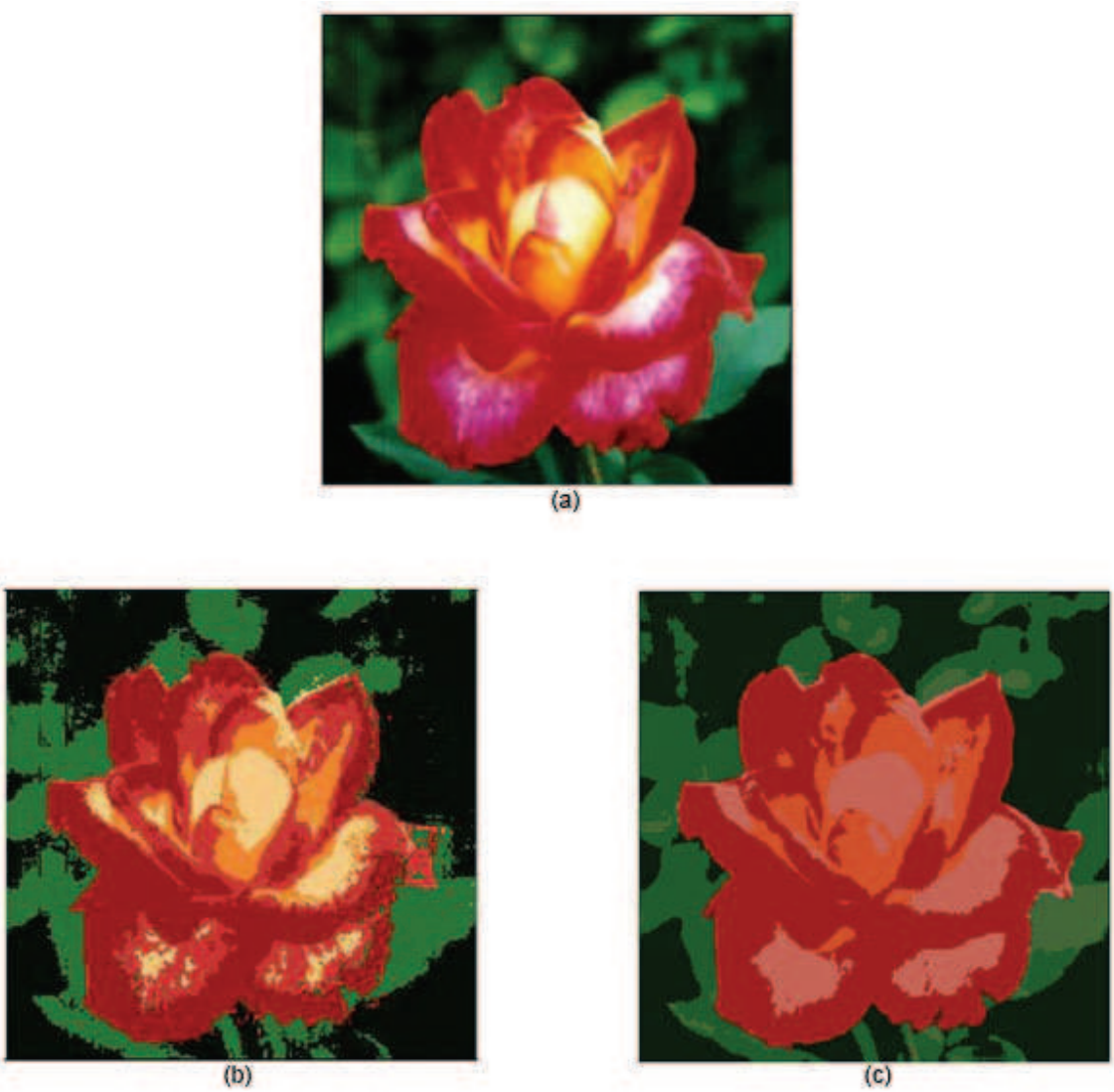

Figure 4.11: (a) The original "rose" image. (b) Segmentation result of the fuzzy homogeneity and SSF approach (six colors). (c) Segmentation using the proposed approach (eight colors).

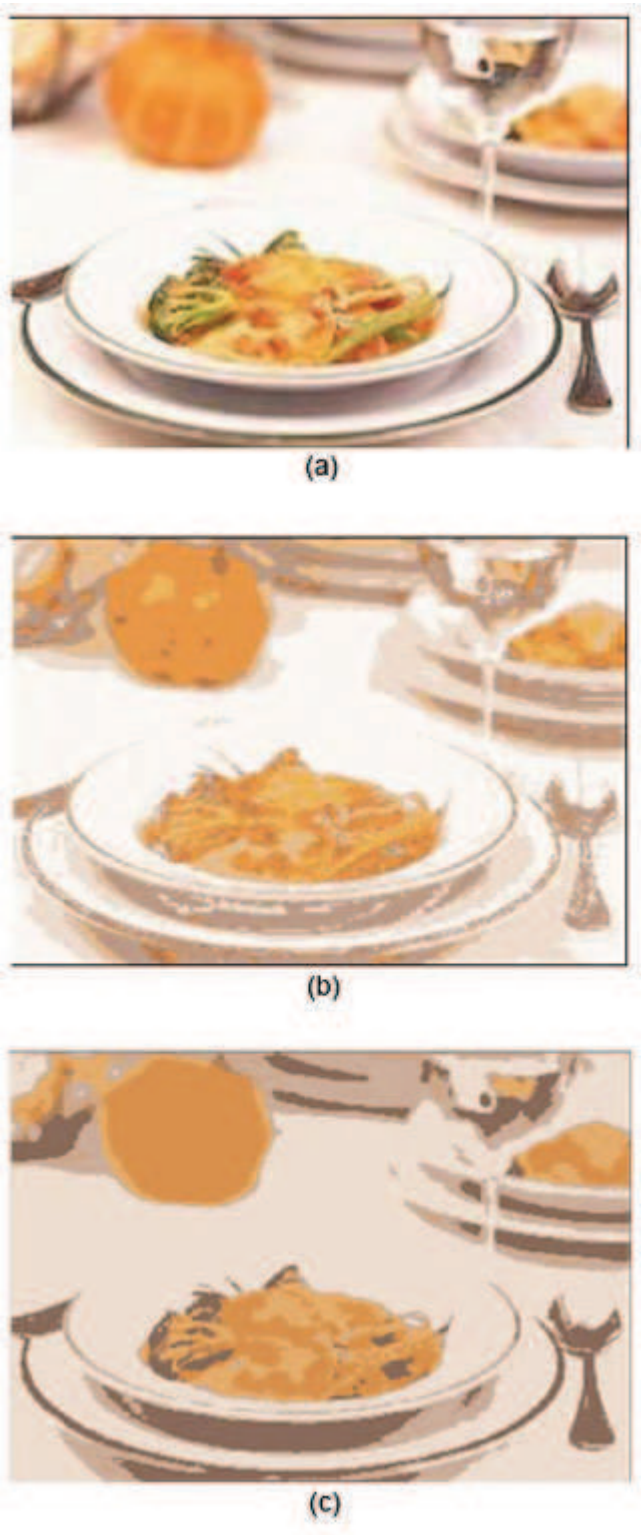

Figure 4.12: (a) The original "salad" image. (b) Segmentation result of the fuzzy homogeneity and SSF approach (eight colors). (c) Segmentation using the proposed approach (seven colors).

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

5.1 Salient Features

The proposed two-stage hierarchical approach based on the self-organizing map, combines the advantages of unsupervised learning based on large datasets and labeling of the clustered outputs. The network detects the dominant color and texture features in a given color image. These are subsequently used to segment the image by pixel classification. Extensive experiments have shown the proposed network to be highly robust and flexible. The dominant features in the image are reassuringly captured, and the final number of clusters formed is adaptively determined. This approach will provides meaningful information by simplifying the image and identifying its constituent regions, thus helping interpretation of natural scene images.

The salient features of the proposed approach can be listed as follows:

- *Coarse Segmentation.* The hierarchical structure of the proposed approach incorporates the concept of coarse segmentation. Experiments have indicated that this step is necessary. The final segmentation is completed based on the result of the coarse segmentation and not by referring to the original image. In fact, direct matching between the pixels of the original image and the very few colors left after color merging would result in a loss of detailed information, leading to poor segmentation quality.

- *Hierarchical Structure.* The hierarchical structure of the approach, which uses a two-dimensional SOM followed by a one-dimensional SOM, is crucial for natural scene segmentation. Experiments have shown that it is very difficult to extract the dominant features accurately if only a one-dimensional SOM is used. In contrast, it is difficult to avoid over-segmentation if only a two-dimensional map is used. Both these problems are overcome in the proposed approach by using the hierarchical framework.

- *Influence of Illumination.* By using the CIE $l^*a^*b^*$ color space, the influence of illumination is reduced significantly. However, there may still be situations in which strongly highlighted or completely shadowed surfaces may lose all their information and the subregions cannot be merged.

- *Comparison.* It is found that any quantitative comparison with segmentation results obtained from other methods is impossible for lack of a satisfactory measure. However, a qualitative comparison made in Section 4.2 does confirm that the proposed approach provides a more intuitive and potentially useful segmentation of natural scenes.

5.2 Future Work

While the following points may not necessarily be huge shortcomings of the proposed approach, they may provide food for thought for future research related to this approach.

- *Removal of misclassified pixels.* The colors of some small but distinct regions are detected in the second stage. Various post-processing methods can be tried out to find one that removes misclassified pixels most effectively.

- *Improvements.* Although the hierarchical network may be improved by a better learning rule during training, there is no appropriate criterion to be optimized for an algorithm to search for a global optimum. However, system performance may be further improved through fine-tuning by the learning vector quantization (LVQ), in which supervised training is required.

- *Utility.* The utility of this approach can be extended by incorporating it as part of larger image processing systems. The capacity for knowledge discovery by such networks has hardly been explored. They could have a huge potential in the field of visual data mining.

REFERENCES

[1] A. Bovik, *Analysisc of multichannel narrow-band filters for image texture segmentation*, IEEE Transactions on Signal Processing, 39 (1991), pp. 2025–2043.

[2] P. Brodatz, *Textures: a photographic album for artists and designers*, Dover Publications, New York, NY, 1966.

[3] J. D. Buf, M. Kardan, and M. Spann, *Texture feature performance for image segmentation*, Pattern Recognition, 23 (1990), pp. 291–309.

[4] R. Chellappa, *Progress in Pattern Recognition*, North-Holland Publ, Amsterdam, 1985, ch. Two dimensional discrete gaussian markov random fields for image processing, pp. 79–112.

[5] P. Chen and T. Pavlidis, *Segmentation by using texture correlation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 5 (1983), pp. 64–69.

[6] H. D. Cheng, X. H. Jiang, Y. Sun, and J. L. Wang, *Color image segmentation: advances and prospects*, Pattern Recognition, 34 (2001), pp. 2259–2281.

[7] H. D. Cheng and J. Li, *Fuzzy homogeneity and scale-space approach to color image segmentation*, Pattern Recognition, 36 (2003), pp. 1545–1562.

[8] H. D. Cheng and Y. Sun, *A hierarchical approach to color image segmentation using homogeneity*, IEEE Transactions on Image Processing, 9 (2000), pp. 2071–2082.

[9] H. D. Cheng, M. Xue, and X. Shi, *Contrast enhancement based on a novel homogeneity measurement*, Pattern Recognition, 36 (2003), pp. 2687–2697.

[10] D. Comaniciu and P. Meer, *Robust analysis of feature spaces: color image segmentation*, in IEEE Conference on Computer Vision and Pattern Recognition, 1997, pp. 750–755.

[11] A. Cumani, *Edge detection in multispectral images*, Computer Vision, Graphics, and Image Processing, 53 (1991), pp. 43–51.

[12] D. Dunn and W. E. Higgins, *Optimal gabor filters for texture segmentation*, IEEE Transactions on Image Processing, 4 (1995), pp. 947–964.

[13] S. Ghosal and R. Mehrotra, *Application of neural networks in segmentation or range images*, in Proceedings of the International Joint Conference on Neural Networks, Baltimore, USA, June 1992, pp. 297–302.

[14] R. Gonzalez and R. Woods, *Digital image processing*, Pearson Education, Upper Saddle River, NJ, 2002.

[15] D. Greenhill and E. R. Davies, *Texture analysis using neural networks and mode filters*, in Proceedings of BMVC93, University of Surrey, Guildford, September 1993, pp. 509–518.

[16] R. Haralick and L. Shapiro, *Survey: Image segmentation techniques*, Computer Vision, Graphics, and Image Processing, 29 (1985), pp. 100–132.

[17] R. M. Haralick, *Statistical and structural approaches to texture*, in Proceedings of the IEEE, vol. 67, May 1979, pp. 786–804.

[18] D. Harwood, M. Subbarao, and L. S. Davis, *Texture classification by local rank corelation*, Computer Vision, Graphics, and Image Processing, 32 (1985), pp. 404–411.

[19] J. Havelicek, *The evolution of modern texture processing*, Elektrik, 5 (1997), pp. 1–28.

[20] X. Huang, Y. Akiri, and M. Jack, *Hidden Markov Models for Speech Recognition*, Edinburgh Univ. Press, Edinburgh, 1990.

[21] T. L. Huntsberger, C. L. Jacobs, and R. L. Cannon, *Iterative fuzzy image segmentation*, Pattern Recognition, 18 (1985), pp. 131–138.

[22] T. L. Huntsberger, C. Rangarajan, and S. N. Jayaramamurthy, *Representation of uncertainty in computer vision using fuzzy sets*, IEEE Transactions on Computers, 35 (1986), pp. 145–156.

[23] A. Jain, *Fundamentals of Digital Image Processing*, Prentice Hall, Englewood Cliffs, NJ, 1989.

[24] A. Jain and F. Farroknia, *Unsupervised texture segmentation using gabr filters*, Pattern Recognition, 24 (1991), pp. 1167–1186.

[25] A. Jain and K. Karu, *Learning texture discrimination masks*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 18 (1996), pp. 195–205.

[26] T. Kohonen, *The self-organizing map*, in Proceedings of the IEEE, vol. 78 of 9, September 1990, pp. 1464–1480.

[27] J. Lampinen and E. Oja, *Clustering properties of hierarchical self-organizing maps*, Journal of Mathematical Imaging and Vision, 2 (1992), pp. 261–272.

[28] K. I. Laws, *Texture energy measures*, in Proccedings of Image Understanding Workshop, November 1979, pp. 47–51.

[29] ——, *Textured image segmentation*, phd thesis, University of Southern California, January 1980.

[30] H.-C. Lee and D. R. Cok, *Detecting boundaries in a vector field*, IEEE Transactions on Signal Processing, 39 (1991), pp. 1181–194.

[31] Y. W. Lim and S. U. Lee, *On the color image segmentation algorithm based on the thresholding and the fuzzy c-means techniques*, Pattern Recognition, 23 (1990), pp. 935–952.

[32] E. Littmann and H. Ritter, *Adaptive color segmentation - a comparison of neural and statistical methods*, IEEE Transactions on Neural Networks, 8 (1997), pp. 175–185.

[33] J. Q. Liu and Y. H. Yang, *Multiresolution color image segmentation*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 16 (1994), pp. 689–700.

[34] P. Miller and S. Astley, *Classification of breast tissue by texture analysis*, in Proceedings of BMVC91, Glasgow, September 1991.

[35] C. Nikias, *Advances in Spectrum Analysis and Array Processing*, Prentice Hall, Englewood Cliffs, New Jersey, 1991, ch. Higher order spectral analysis, pp. 326–365.

[36] R. Ohlander, K. Price, and D. R. Pierson, *Picture segmentation using a recursive region splitting method*, Computer Graphics and Image Processing, 8 (1978), pp. 313–333.

[37] R. Ohlander, K. Price, and D. R. Reddy, *Picture segmentation using recursuve region splitting method*, Computer Graphics and Image Processing, 8 (1978), pp. 313–333.

[38] Y.-I. Ohta and T. Kanade, *Color information for region segmentation*, Computer Graphics and Image Processing, 13 (1980), pp. 222–241.

[39] M. T. Orchard and C. A. Bouman, *Color quantization of images*, IEEE Transactions on Signal Processing, 39 (1991), pp. 2677–2690.

[40] N. R. Pal and S. K. Pal, *A review on image segmentation techniques*, Pattern Recognition, 26 (1993), pp. 1277–1294.

[41] N. Papamarkos, *Color reduction using local features and a sofm neural network*, International Journal Imaging Systems and Technology, 10 (1999), pp. 404–409.

[42] M. Pietikainen and et. al., *Accurate color discrimination with classification based on feature distributions*, in International Conference on Pattern Recognition, vol. C, 1996, pp. 833–838.

[43] M. Pietikainen, A. Rosenfeld, and L. S. Davis, *Texture classification using averages of local pattern matches*, in Proceedings of IEEE Computer Society Conference on Pattern Recognition and Image Processing, 1982.

[44] ——, *Experiments with texture classification using averages of local pattern matches*, IEEE Transactions on Systems, Man and Cybernetics, SMC-13 (1983), pp. 421–426.

[45] A. Rao and B. Schunk, *Computing oriented texture fields*, Computer Vision, Graphics, and Image Processing, 52 (1991), pp. 157–185.

[46] H. Traven, *A neural network clustering algorithm, and its application to multispectral satellite image classification*, in Proceedings of the Sixth Scandinavian Conference on Image analysis, Oulu, Finland, June 1989, pp. 128–135.

[47] M. R. Turner, *Texture discrimination by gabor functions*, Biologica Cybernetics, 55 (1986), pp. 71–82.

[48] T. P. Weldon and W. E. Higgins, *An algorithm for designing multiple gabor filters for segmenting multi-textured images*, in Proceedings of the IEEE International Conference on Image Processing, vol. 3, Chicago, Illinois, October 4-7 1998, pp. 333–337.