

**SEED PRUNING FOR MULTI-RESOLUTION SEGMENTATION OF  
VASCULATURE IN IMMUNOHISTOCHEMICAL IMAGES**

by

Rohit Chacko Philip

Department of Electrical and Computer Engineering

University of Arizona

A Thesis Submitted to the Faculty of the  
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

In Partial Fulfillment of the Requirements  
For the Degree of

MASTER OF SCIENCE  
WITH A MAJOR IN ELECTRICAL AND COMPUTER ENGINEERING

In the Graduate College

THE UNIVERSITY OF ARIZONA

2008

### STATEMENT BY AUTHOR

This thesis has been submitted in partial fulfillment of requirements for an advanced degree at The University of Arizona and is deposited in the University Library to be made available to borrowers under rules of the Library.

Brief quotations from this thesis are allowable without special permission, provided that accurate acknowledgment of source is made. Requests for permission for extended quotation from or reproduction of this manuscript in whole or in part may be granted by the head of the major department or the Dean of the Graduate College when in his or her judgment the proposed use of the material is in the interests of scholarship. In all other instances, however, permission must be obtained from the author.

SIGNED: \_\_\_\_\_

### APPROVAL BY THESIS DIRECTORS

This thesis has been approved on the date shown below:

\_\_\_\_\_  
Jeffrey J. Rodriguez  
Associate Professor of  
Electrical and Computer  
Engineering.

\_\_\_\_\_  
Robert J. Gillies  
Professor of Radiology,  
Biochemistry and Molecular  
Biophysics.

## Acknowledgements

I would like to express my heartfelt thanks to Dr. Jeffrey J. Rodríguez, for his guidance throughout the course of this research project. I would also like to thank him for his support and motivation throughout my period of study at The University of Arizona.

I am also extremely grateful to Dr. Robert J. Gillies, for providing me the opportunity to work on this research project. It has been a very challenging research project, the completion of which would not have been possible without his ideas, suggestions and continued support.

I would also like to express my thanks to Dr. Ivan B. Djordjevic, for agreeing to be on my thesis committee, and taking the time to read my thesis.

I would like to thank Dr. Ronald M. Lynch, for giving me the opportunity to work with the Arizona Cancer Center Core Imaging Group, which has contributed immensely to the growth of my knowledge base in my specialized field of study.

I would also like to thank Dr. Natarajan Raghunand, for all his help and suggestions throughout my work.

I would like to thank Mr. James Averill and Ms. Kathy McDaniel at the TACMASS lab at the Arizona Cancer Center, for providing me with valuable image datasets for testing.

I would like to especially thank Ms. Merry Warner, for all the resources and motivation extended towards my work.

I would also like to especially thank Ms. Tami Whelan, for all the efficient management of my study as an international student at the Electrical and Computer Engineering Department of The University of Arizona.

I would like to thank my parents Dr. Philip Chacko and Dr. Shirley Philip, and my cousin Nithya Abraham for their love, support and encouragement throughout my life.

I would like to thank my colleagues and friends in the Signal and Imaging Laboratory, especially Sivaramakrishnan Rajaraman and Narasimhan Rajagopalan, who have provided me with valuable inputs towards my thesis project.

My thanks also go out to all my friends, who have always been there for me, especially Somasundaram Perianayagam and Dheepan Shanmugasundaram for their love and support throughout my stay at Tucson.

*To my Parents*

## TABLE OF CONTENTS

<b>LIST OF FIGURES .....</b>	<b>7</b>
<b>LIST OF FIGURES ~ <i>CONTINUED</i> .....</b>	<b>8</b>
<b>LIST OF TABLES .....</b>	<b>9</b>
<b>ABSTRACT.....</b>	<b>10</b>
<b>1. INTRODUCTION .....</b>	<b>11</b>
1.1 Breast Cancer .....	11
1.2 Project Goals .....	12
1.3 Image Acquisition .....	12
1.4 Immunohistochemistry .....	13
<b>2. PRELIMINARY WORK .....</b>	<b>18</b>
2.1 Color Profiling .....	20
<b>3. COLOR SPACES .....</b>	<b>24</b>
3.1 <i>RGB</i> Color Space .....	24
3.2 <i>rgb</i> Color Space .....	26
3.3 <i>XYZ</i> Color Space .....	27
3.4 <i>xyz</i> Color Space .....	28
3.5 <i>YUV</i> Color Space .....	29
3.6 <i>YIQ</i> Color Space .....	30
3.7 <i>YCbCr</i> Color Space .....	32
3.8 <i>Opponent</i> Color Space .....	33
3.9 <i>OhtaI<sub>1</sub>I<sub>2</sub>I<sub>3</sub></i> Color Space .....	34
3.10 <i>IHS</i> Color Space .....	35
3.11 <i>HSV</i> Color Space .....	37
3.12 <i>HLS</i> Color Space .....	38
3.13 <i>CIELAB</i> Color Space .....	39
3.14 <i>CIELUV</i> Color Space .....	40
<b>4. IMAGE SEGMENTATION SCHEMES.....</b>	<b>43</b>
4.1 Thresholding techniques .....	43
4.2 Boundary-based techniques .....	44
4.3 Region-based techniques .....	44
4.4 Hybrid techniques .....	45
4.5 3-D Histogram method .....	45
4.6 Pulse Coupled Neural Network method (PCNN) .....	46
4.7 k-means Clustering method .....	50
4.8 Integrated Color Edges method .....	52
4.9 Histogram Peak Selection method .....	53

## TABLE OF CONTENTS ~ *Continued*

<b>5. PRINCIPAL COMPONENT ANALYSIS.....</b>	<b>56</b>
<b>6. SEED PRUNING ALGORITHM.....</b>	<b>64</b>
<b>7. REGION GROWING.....</b>	<b>68</b>
<b>8. RESULTS – SEGMENTATED IMAGES.....</b>	<b>74</b>
<b>9. RESULTS – ANALYSIS AND DISCUSSION.....</b>	<b>80</b>
9.1 Segmentation Accuracy .....	80
9.2 Efficiency in reducing number of seed pixels.....	81
9.3 Effect of number of resolution levels on efficiency and accuracy.....	82
9.4 False Negative Ratio and False Positive Ratio .....	83
9.5 Region of Convergence curves based on varying thresholds .....	84
<b>10. IMAGE REGISTRATION .....</b>	<b>91</b>
10.1 Cross Correlation .....	93
<b>11. 3-D MODEL CREATION.....</b>	<b>97</b>
<b>12. CONTRIBUTIONS AND FUTURE WORK.....</b>	<b>100</b>
12.1 Contributions.....	100
12.2 Future Work .....	102
<b>13. REFERENCES.....</b>	<b>103</b>

## List of Figures

Figure 1: Dataset 1 showing ROI.....	18
Figure 2: Dataset 2 showing ROIs.....	19
Figure 3: Dataset 2 Image showing ROIs.....	21
Figure 4: Segmentation using color profiling.....	21
Figure 5: Dataset 2 Image showing ROIS.....	22
Figure 6: Segmentation using color profiling.....	22
Figure 7: Image showing ROI in <i>RGB</i> color space.....	25
Figure 8: Image showing ROI in <i>rgb</i> color space.....	26
Figure 9: Image showing ROI in <i>XYZ</i> color space.....	27
Figure 10: Image showing ROI in <i>xyz</i> color space.....	28
Figure 11: Image showing ROI in <i>YUV</i> color space.....	29
Figure 12: Image showing ROI in <i>YUV (IHS type)</i> color space.....	30
Figure 13: Image showing ROI in <i>YIQ</i> color space.....	31
Figure 14: Image showing ROI in <i>YIQ (IHS type)</i> color space.....	31
Figure 15: Image showing ROI in <i>YC<sub>b</sub>C<sub>r</sub></i> color space.....	32
Figure 16: Image showing ROI in <i>Opponent</i> color space.....	34
Figure 17: Image showing ROI in <i>Ohta</i> color space.....	35
Figure 18: Image showing ROI in <i>IHS</i> color space.....	36
Figure 19: Image showing ROI in <i>HSV</i> color space.....	37
Figure 20: Image showing ROI in <i>HLS</i> color space.....	39
Figure 21: Image showing ROI in <i>CIELAB</i> color space.....	40
Figure 22: Image showing ROI in <i>CIELUV</i> color space.....	41
Figure 23: A k-means cluster showing 12 clusters.....	51
Figure 24: A thresholded k-means cluster segmentation.....	52
Figure 25: Flowchart showing the working of the Seed Pruning Algorithm with number of each class of pixels for a specific case highlighted at each step.....	65
Figure 26: 8-connected neighbor pixels of pixel <i>x</i> .....	70
Figure 27: Automated segmentation using our approach.....	74
Figure 28: Manual segmentation.....	74
Figure 29: Automated segmentation using our approach.....	75
Figure 30: Manual segmentation.....	75
Figure 31: Automated segmentation using our approach.....	76
Figure 32: Manual segmentation.....	76
Figure 33: Automated segmentation using our approach.....	77
Figure 34: Manual segmentation.....	77
Figure 35: Automated segmentation using our approach.....	78
Figure 36: Manual segmentation.....	78
Figure 37: Automated segmentation using our approach.....	79
Figure 38: Manual segmentation.....	79
Figure 39: Number of <i>Final seed pixels</i> Vs. <i>Candidate seed pixel</i> selection threshold. ..	86
Figure 40: Accuracy, FNR and FPR Vs. <i>Candidate seed pixel</i> selection threshold.....	86

### List of Figures ~ *Continued*

Figure 41: Number of <i>Final seed pixels</i> Vs. <i>Candidate seed pixel</i> neighbor percentage. 87	87
Figure 42: Accuracy, FNR and FPR Vs. <i>Candidate seed pixel</i> neighbor percentage. .... 87	87
Figure 43: Accuracy, FNR and FPR Vs. Homogeneity criterion. .... 89	89
Figure 44: Accuracy, FNR and FPR Vs. Closing Window size. .... 90	90
Figure 45: Reference Image. .... 95	95
Figure 46: Sensed Image before registration. .... 95	95
Figure 47: Sensed Image registered to Reference Image. .... 96	96
Figure 48: 3-D model of segmented data viewed using Slicer Dicer. .... 98	98
Figure 49: Oblique slice passing through an oblique plane, showing data components. . 99	99



## List of Tables

Table 1: NUMBER OF SEED PIXELS AND SEGMENTATION ACCURACY .....	81
Table 2: NUMBER OF SEED PIXELS AND SEGMENTATION ACCURACY OF THE SEED PRUNING METHOD BY VARYING RESOLUTION LEVELS .....	82
Table 3: FALSE NEGATIVE AND FALSE POSITIVE RATIOS .....	84

## **ABSTRACT**

An automated system for the segmentation of vasculature in high-resolution digital color images of immunohistochemically stained slides of breast cancer tissue is presented. Manual immunohistochemical staining processes result in differences in stain retention by the regions of interest (ROIs). Difficulties in the segmentation processes arise when the stain retention is poor, causing the existing color profiling schemes in the literature to fail. The automated segmentation algorithm uses a principal component region-growing scheme with automated generation of seed pixels. The number of candidate seed pixels is extremely large due to the high-resolution. The main focus of the research is to present a multi-resolution scheme for accurate selection of a minimal number of seed pixels to be presented as inputs to the region-growing segmentation algorithm. The system is tested for accuracy, and the efficiency is measured in terms of percentage reduction in number of seed pixels, as well as accuracy of the segmentation results and false positive and false negative pixel classification ratios. The effect of the number of resolution levels on the accuracy and number of seed pixels selected is also discussed.

## **Chapter 1**

### **1. INTRODUCTION**

#### **1.1 Breast Cancer**

Breast cancer is a cancer that starts in the cells of the breast. Worldwide, breast cancer is the second most common type of cancer after lung cancer (10.4% of all cancer incidence, both sexes counted) and the fifth most common cause of cancer death. Worldwide, breast cancer is by far the most common cancer amongst women, with an incidence rate more than twice that of colorectal cancer and cervical cancer and about three times that of lung cancer. However breast cancer mortality worldwide is just 25% greater than that of lung cancer in women. In 2005, breast cancer caused 502,000 deaths worldwide (7% of cancer deaths; almost 1% of all deaths). The number of cases worldwide has significantly increased since the 1970s, a phenomenon partly blamed on modern lifestyles in the Western world.

In the US, both incidence and death rates for breast cancer have been declining in the last few years. Nevertheless, a US study conducted in 2005 by the Society for Women's Health Research indicated that breast cancer remains the most feared disease, even though heart disease is a much more common cause of death among women.

## **1.2 Project Goals**

This project is aimed at producing high quality 3-D models of the blood vasculature and tumors present in breast cancer tissue in order to aid breast cancer research. The primary research goal of the project is to segment the regions of interest (hereafter referred as ROIs) which comprise of blood vasculature and/or tumors from the general tissue image comprising of additional cellular components like fats, non-cancerous tissue etc.

## **1.3 Image Acquisition**

The Tissue Acquisition and Cellular/Molecular Analysis Shared Service (TACMASS) provide support to both the Arizona Cancer Center (AZCC) and The University of Arizona (UA) research investigators. TACMASS offers services for routine histology using both frozen and formalin fixed, paraffin embedded (FFPE) human and research animal tissues and cell lines. Immunohistochemistry (IHC) and in situ hybridization are performed on an automated platform (Ventana Discovery XT).

The breast cancer tissue slices are stained with a coloring agent (Cd-31) and are imaged using a DMetrix array microscope. The DMetrix system is distinguished by its virtual slide processing rate. One complete, high-resolution image file of a microscope slide can exceed 10 gigabytes in size. DMetrix's patent-pending digital-pathology imaging system is based on a revolutionary light microscope technology: 80 tiny

microscopes are arranged in staggered rows as a compact array. The entire array measures less than 1 inch across and less than one-third inch in height. The array glides along the surface of a glass slide to capture its images. This technology has been named "array microscopy." Fabrication of the device leverages recent advances in miniaturization and ultra-precise manufacturing.

As a result, the images under consideration are high-resolution digital color images of histological slides, containing breast cancer tissue slices stained using immunohistochemistry (IHC) techniques, with an average file size of 1,066,048 Kb each, having a pixel size of  $0.5 \mu\text{m} \times 0.5 \mu\text{m}$  with image dimensions of  $16512 \mu\text{m} \times 27168 \mu\text{m}$  ( $33024 \times 54336$  pixels) corresponding to 20X magnification on an average.

## **1.4 Immunohistochemistry**

Immunohistochemistry or IHC refers to the process of localizing proteins in cells of a tissue section exploiting the principle of antibodies binding specifically to antigens in biological tissues. It takes its name from the roots "immuno," in reference to antibodies used in the procedure, and "histo" meaning tissue (compare to immunocytochemistry). Immunohistochemical staining is widely used in the diagnosis of abnormal cells such as those found in cancerous tumors. Specific molecular markers are characteristic of particular cellular events such as proliferation or cell death (apoptosis). IHC is also widely used in basic research to understand the distribution and localization of biomarkers and differentially expressed proteins in different parts of a biological tissue.

Visualizing an antibody-antigen interaction can be accomplished in a number of ways. In the most common instance, an antibody is conjugated to an enzyme, such as peroxidase, that can catalyze a color-producing reaction. Alternatively, the antibody can also be tagged to a fluorophore, such as FITC, rhodamine, Texas Red, Alexa Fluor, or DyLight Fluor. The latter method is of great use in confocal laser scanning microscopy, which is highly sensitive and can also be used to visualize interactions between multiple proteins.

IHC is an excellent detection technique and has the tremendous advantage of being able to show exactly where a given protein is located within the tissue examined. This has made it a widely-used technique in the neurosciences, enabling researchers to examine protein expression within specific brain structures. Its major disadvantage is that, unlike immunoblotting techniques where staining is checked against a molecular weight ladder, it is impossible to show in IHC that the staining corresponds with the protein of interest. For this reason, primary antibodies must be well-validated in a Western Blot or similar procedure. The technique is even more widely used in diagnostic surgical pathology for typing tumors (e.g. carcinoma vs. melanoma).

### **1.4.1 Antibody Types**

The antibodies used for specific detection can be polyclonal or monoclonal. Monoclonal antibodies are generally considered to exhibit greater specificity. Polyclonal antibodies are made by injecting animals with peptide antigens, and then after a secondary immune response are stimulated, isolating antibodies from whole serum. Thus,

polyclonal antibodies are a heterogeneous mix of antibodies that recognize several epitopes.

Antibodies can also be classified as primary or secondary reagents. Primary antibodies are raised against an antigen of interest and are typically unconjugated (unlabelled), while secondary antibodies are raised against primary antibodies. Hence, secondary antibodies recognize immunoglobulin of a particular species and are conjugated to either biotin or a reporter enzyme such as alkaline phosphatase or horseradish peroxidase. Some secondary antibodies are conjugated to fluorescent agents, such as the Alexa Fluor or Dylight Fluor family, are also frequently used for detection of proteins in IHC procedures. Protein concentration is generally measured by densitometry analysis, where the intensity of staining correlates with the amount of the protein of interest.

### **1.4.2 Sample Preparation**

In the procedure, depending on the purpose and the thickness of the experimental sample, either thin (about 4-40  $\mu\text{m}$ ) slices are taken of the tissue of interest, or if the tissue is not very thick and is penetrable it is used whole. The slicing is usually accomplished through the use of a microtome, and slices are mounted on slides. "Free-floating IHC" uses slices that are not mounted; these slices are normally produced using a vibrating microtome.

### 1.4.3 Direct and Indirect IHC

There are two strategies used for the immunohistochemical detection of antigens in tissue, the *direct* method and the *indirect* method. In both cases, the tissue is treated to rupture the membranes, usually by using a kind of detergent such as Triton X-100. Some antigen also need additional step for unmasking, resulting in better detection results.

The *direct method* is a one-step staining method, and involves a labeled antibody (e.g. FITC conjugated antiserum) reacting directly with the antigen in tissue sections. This technique utilizes only one antibody and the procedure is therefore simple and rapid. However, it can suffer problems with sensitivity due to little signal amplification and is in less common use than indirect methods.

The indirect method of immunohistochemical staining uses one antibody against the antigen being probed for, and a second, labelled, antibody against the first.

The *indirect method* involves an unlabeled primary antibody (first layer) which reacts with tissue antigen, and a labeled secondary antibody (second layer) which reacts with the primary antibody. (The secondary antibody must be against the IgG of the animal species in which the primary antibody has been raised.) This method is more sensitive due to signal amplification through several secondary antibody reactions with different antigenic sites on the primary antibody. The second layer antibody can be labeled with a fluorescent dye or an enzyme.

In a common procedure, a biotinylated secondary antibody is coupled with streptavidin-horseradish peroxidase. This is reacted with 3, 3'-Diaminobenzidine (DAB)



to produce a brown staining wherever primary and secondary antibodies are attached in a process known as DAB staining. The reaction can be enhanced using nickel, producing a deep purple/gray staining.

The indirect method, aside from its greater sensitivity, also has the advantage that only a relatively small number of standard conjugated (labeled) secondary antibodies need to be generated. For example, a labeled secondary antibody raised against rabbit IgG, which can be purchased "off the shelf," is useful with any primary antibody raised in rabbit. With the direct method, it would be necessary to make custom labeled antibodies against every antigen of interest.

## Chapter 2

### 2. PRELIMINARY WORK

This chapter deals with the initial work done in the preliminary stages of research that sets up the thesis problem.

The images under consideration for the project was a set of eight Cd-31 stained images (dataset 1), followed by another set of forty Cd-31 stained images (dataset 2). The difference in the two datasets is that they were both obtained at different times.

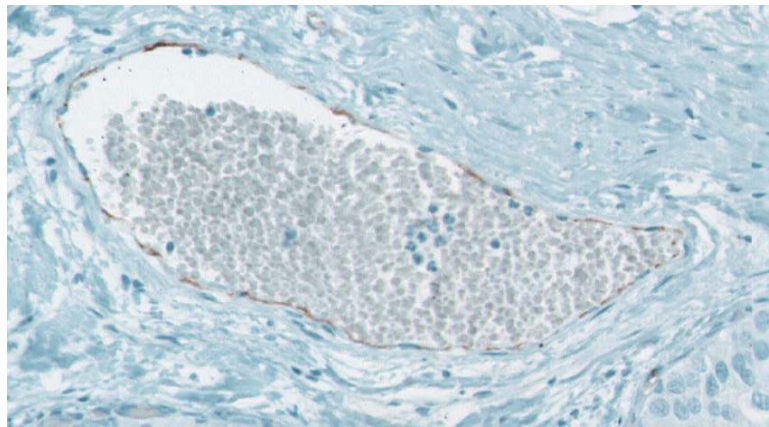


Figure 1: Dataset 1 showing ROI.

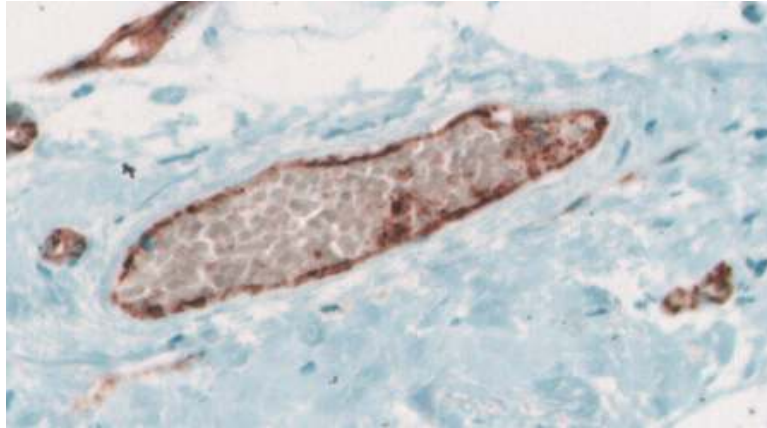


Figure 2: Dataset 2 showing ROIs.

The primary standout feature between the ROIs and the background in both datasets is the color information. This falls in line with the staining process because the tissue is stained with a coloring agent so that the ROIs retain the stain and when imaged they show a different color with respect to the background.

Most manual immunohistochemical staining methods results in the antibody (ROI) reacting with the protein to produce a different color as compared to the rest of the background tissue. The amount of color retention by the ROIs also depends on the amount of time that is elapsed between staining and imaging. As a result different staining runs produce slight differences in stain retention by the ROIs and consequently the two datasets are different in the sense that the ROIs in them retain different levels of the coloring stain, and appear to be of different color.

## 2.1 Color Profiling

Duffy et al. [1] have explained a method for object detection using colour, using a colour profile and colour histograms . Colour profiling learns to detect objects which are members of a particular class, a class being a set of objects that share similar or overlapping colour distributions.

Two sets of training images are used, one set is of images known to contain at least one target object, and the other is a set of images free of the target object. In the training images with target objects, the location and size of the target objects are not known, nor is the exact number of target objects present.

A small change to these algorithms that profile the target object color from a set of training images is used here. The ROI in a training image is manually segmented and the color is profiled by finding the mean and standard deviations of the R, G and B components in the manually segmented piece. The original image is then segmented by matching all regions whose R, G, *and* B values fall within one standard deviation point on either side of the mean value of the color learnt by the profiling technique.

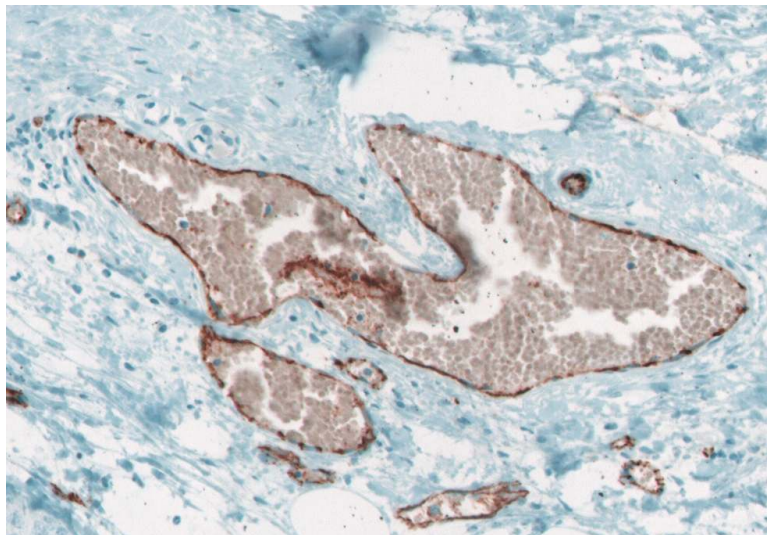


Figure 3: Dataset 2 Image showing ROIs.

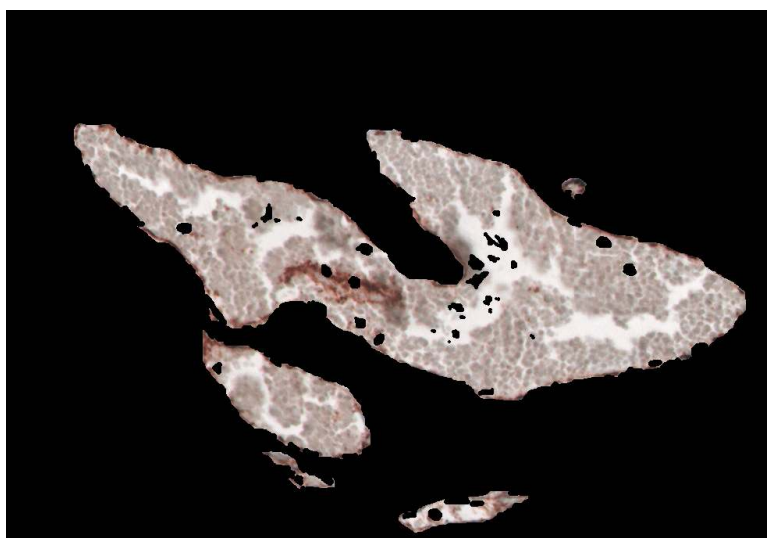


Figure 4: Segmentation using color profiling.

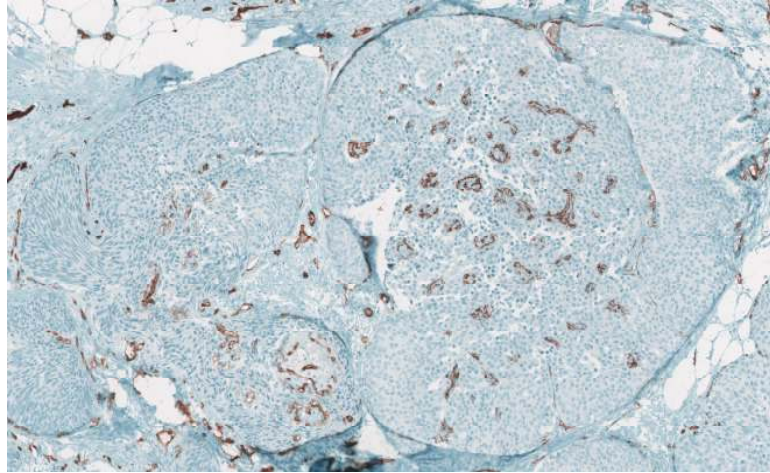


Figure 5: Dataset 2 Image showing ROIS.

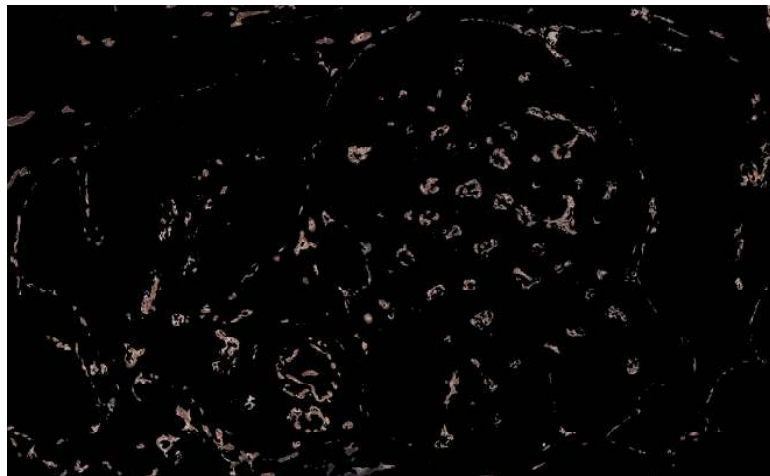


Figure 6: Segmentation using color profiling.

As discussed earlier, different staining runs result in differences in retention of the coloring stain by the ROIs. Therefore the two datasets are vastly different from each other in terms of color profile. The ROIs are not as easily distinguishable from the background in dataset 1, as they were in dataset 2. This causes the color profiling method

to fail when applied to dataset 1. An alternative method is needed to segment the ROIs in dataset 1.

Thus, we arrive at the **problem definition** for the thesis - The research into segmentation schemes that segment the ROIs (blood vasculature) from high-resolution immunohistochemical images, which are poorly stained due to poor stain (color) retention by the ROIs and/or differences in manual staining methods which cause poorer stain retention, where the color profiling technique in the literature fails to provide acceptable segmentations.

## Chapter 3

### 3. COLOR SPACES

This chapter discusses the various color spaces that the RGB color image can be transformed into for color profiling to work.

The digital images of the high-resolution histological slides produced by Cd-31 staining of breast cancer tissue, is a three channel RGB (red, green, blue) digital image. Since dataset 1. did not produce acceptable segmentations with the color profile method that worked quite acceptably for dataset 2, there was a need to convert the image from the RGB color space into various other color spaces in order to test the working of the color profiling algorithm.

Color spaces (also known as color coordinate systems, or color models) are three-dimensional arrangements of color sensations [2]. Colors are specified by points in these spaces. The color spaces presented in this chapter are the most popular in the image processing community. Equations describing transformations between different colorspace are presented. Spaces used in image processing are derived from visual system models (e.g. RGB, Opponent, IHS etc.); adopted from technical domains (e.g. XYZ, YUV etc.) or developed especially for image processing (e.g. Ohta).

#### 3.1 *RGB* Color Space



The *RGB* space [2] is the most frequently used color space for image processing. Since image acquisition systems and image display systems are most often provided with a direct *RGB* signal input or output, this color space is the basic one, which is, if necessary transformed into other color spaces. However the *RGB* primaries of these devices are not always consistent. The color gamut in *RGB* space forms a cube. Each color, which is described by its *RGB* components, is represented by a point and can be found either on the surface or inside the cube. All grey colors are placed on the main diagonal of this cube from black ( $R = G = B = 0$ ) to white ( $R = G = B = \text{max}$ ).

The main disadvantage of *RGB* color space in applications involving natural images is a high correlation [2] between its components: about 0.78 for  $B - R$ , 0.98 for  $R - G$  and 0.94 for  $G - B$ , components. This makes the *RGB* space unsuitable for compression. Another disadvantage of *RGB* color space is non-uniformity, i.e. it is impossible to evaluate the perceived differences between colors on the basis of distance in *RGB* space.

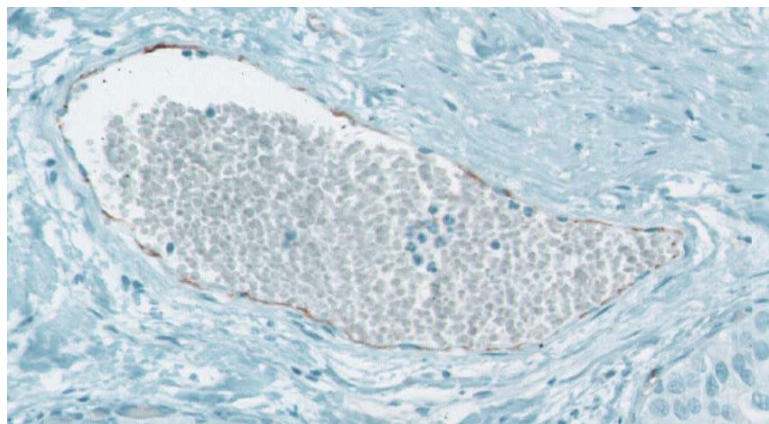


Figure 7: Image showing ROI in *RGB* color space.

### 3.2 *rgb* Color Space

The *RGB* components for a given image are proportional to the amount of light incident on the scene represented by the image. In order to eliminate the influence of illumination intensity, so-called chromaticity coordinates (normalized colors) were introduced in colorimetry [2].

$$r = \frac{R}{R+G+B} \quad (3.2.1)$$

$$g = \frac{G}{R+G+B} \quad (3.2.2)$$

$$b = \frac{B}{R+G+B} = 1 - r - g \quad (3.2.3)$$

Since each of the normalized colors is linearly dependent, the *rgb* space may be represented by two normalized colors. Values of *rgb* coordinates are much more stable with changes in illumination [2].

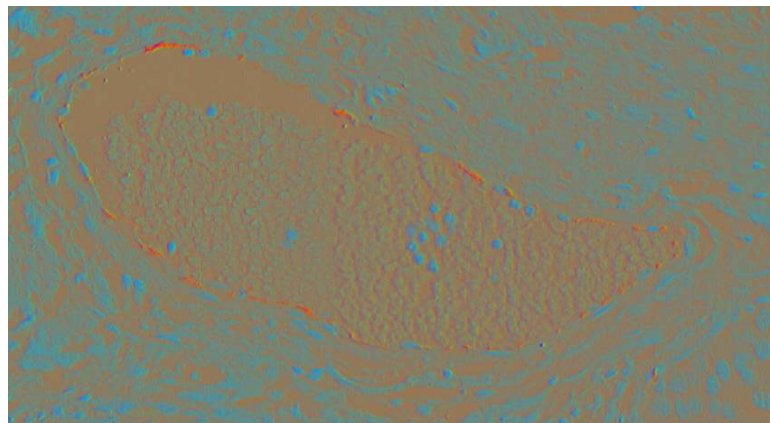


Figure 8: Image showing ROI in *rgb* color space.

### 3.3 XYZ Color Space

The *XYZ* color space [2] was designed to yield non-negative tristimulus values for each color. In this system, *Y* represents the luminance of the color. The tristimulus values *XYZ* are related to *CIE RGB* tristimulus values by the following equations.

$$X = 0.490R + 0.310G + 0.200B \quad (3.3.1)$$

$$Y = 0.177R + 0.812G + 0.011B \quad (3.3.2)$$

$$Z = 0.000R + 0.010G + 0.090B \quad (3.3.3)$$

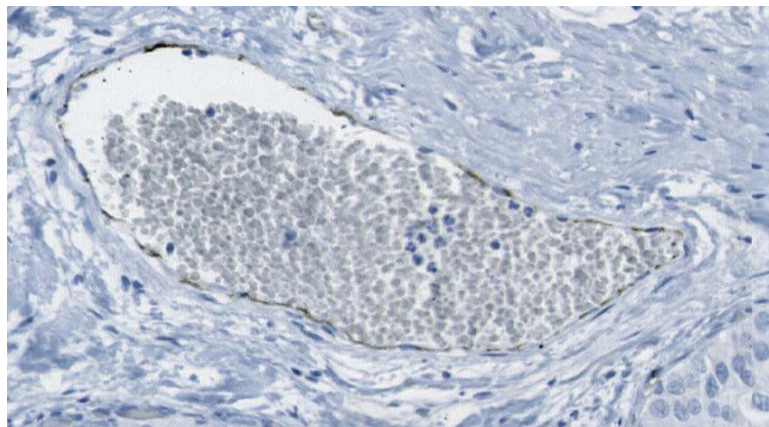


Figure 9: Image showing ROI in *XYZ* color space.

### 3.4 xyz Color Space

In colorimetry the following transformation equations for chromaticity coordinates  $x$ ,  $y$ ,  $z$  were used. This resulted in the transformation into a  $xyz$  color space [2].

$$x = \frac{X}{X+Y+Z} \quad (3.4.1)$$

$$y = \frac{Y}{X+Y+Z} \quad (3.4.2)$$

$$z = \frac{Z}{X+Y+Z} = 1 - x - y \quad (3.4.3)$$

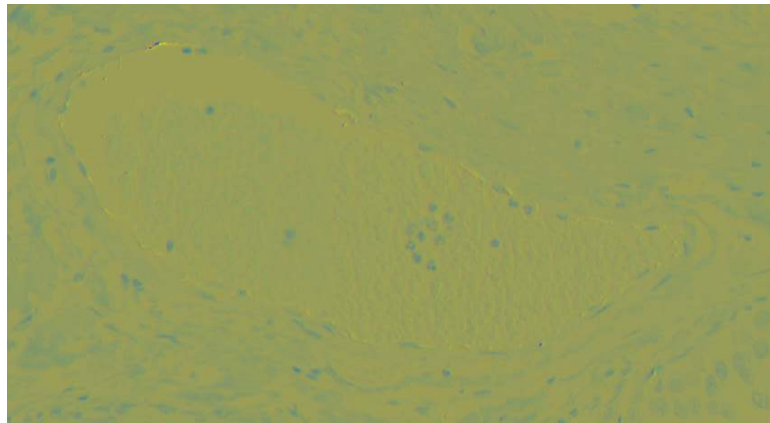


Figure 10: Image showing ROI in  $xyz$  color space.

### 3.5 YUV Color Space

The  $Y$  (luminance) component is identical to the  $Y$  component in  $XYZ$  space. The  $UV$  plane is widely used in coding of color images in color video. Because the color representation of an image sequence requires less detail than the luminance, the data rate can be shared as follows 80% to the  $Y$  component and 10% each to the  $U$  and  $V$  components [2].

$$Y = 0.299R + 0.587G + 0.114B \quad (3.5.1)$$

$$U = -0.147R - 0.289G + 0.437B = 0.493(B - Y) \quad (3.5.2)$$

$$V = 0.615R - 0.515G - 0.100B = 0.877(R - Y) \quad (3.5.3)$$

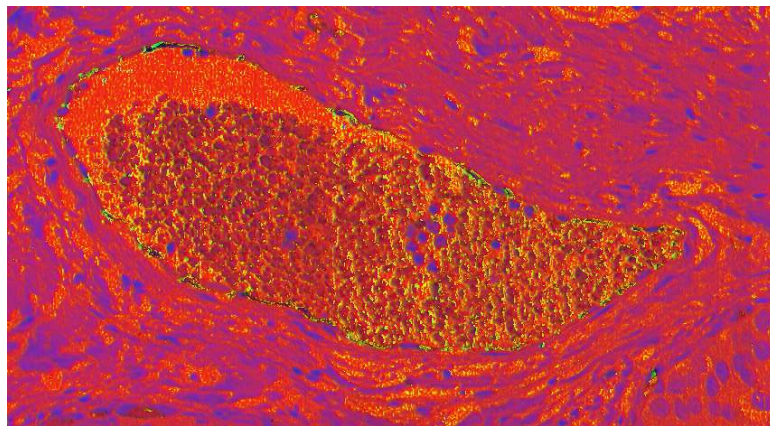


Figure 11: Image showing ROI in  $YUV$  color space.

The  $YUV$  space can be transformed into an  $IHS$ -type space.

$$H_{UV} = \tan^{-1}\left(\frac{V}{U}\right) \quad (3.5.4)$$

$$S_{UV} = \sqrt{(U^2 + V^2)} \quad (3.5.5)$$

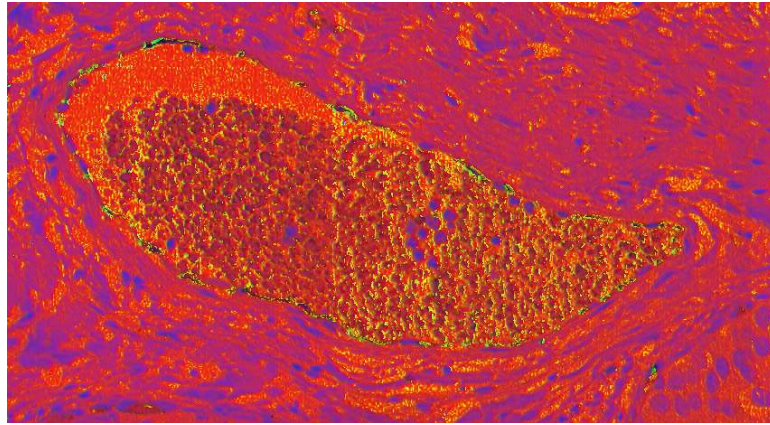


Figure 12: Image showing ROI in  $YUV$  ( $IHS$  type) color space.

### 3.6 $YIQ$ Color Space

As with the  $YUV$  space [2], the  $Y$  (luminance) component is identical to  $Y$  in  $XYZ$  space, while the  $I$  component (in-phase, an orange-cyan axis) and the  $Q$  component (quadrature, a magenta-green axis) express jointly hue and saturation [2].

$$Y = 0.299R + 0.587G + 0.114B \quad (3.6.1)$$

$$I = 0.596R - 0.274G - 0.322B = 0.74(R - Y) - 0.27(B - Y) \quad (3.6.2)$$

$$Q = 0.211R - 0.523G + 0.312B = 0.48(R - Y) + 0.41(B - Y) \quad (3.6.3)$$

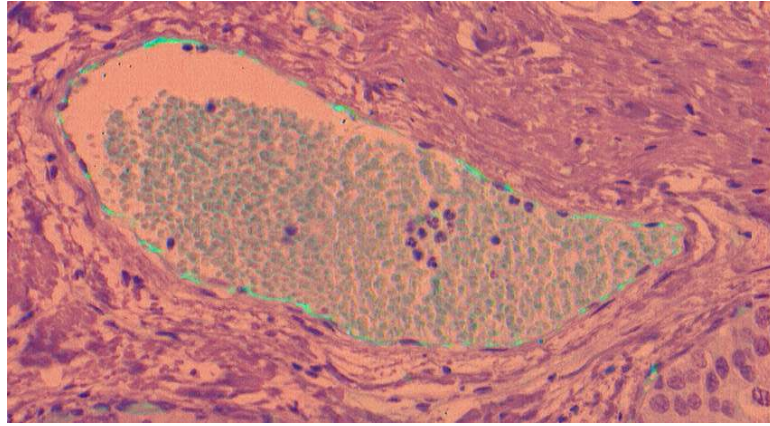


Figure 13: Image showing ROI in  $YIQ$  color space.

The YUV space can be transformed into an IHS-type space.

$$H_{IQ} = \tan^{-1}\left(\frac{Q}{I}\right) \quad (3.6.4)$$

$$S_{IQ} = \sqrt{(I^2 + Q^2)} \quad (3.6.5)$$

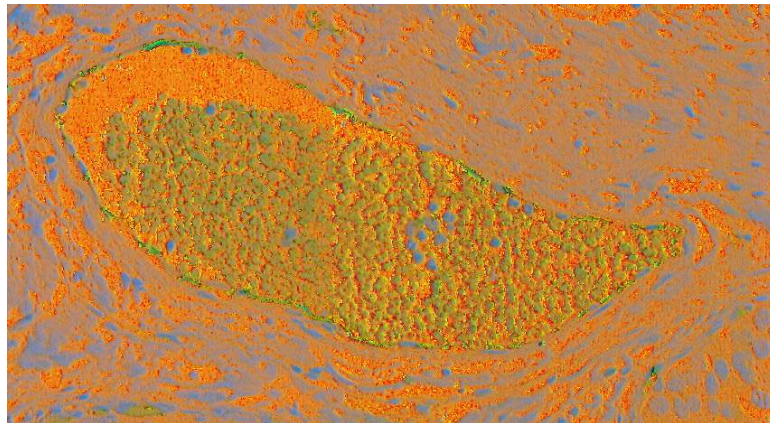


Figure 14: Image showing ROI in  $YIQ$  (*IHS type*) color space.

### 3.7 $YC_bC_r$ Color Space

The  $Y$  component is identical to that for  $YUV$  and  $YIQ$ . Current applications in image compression (e.g. JPEG format) often employ  $YC_bC_r$  space as a quantization space [2]. Greater compression is achieved using reduced spatial resolution and coarser quantization for  $C_b$  and  $C_r$  than for the  $Y$  component.

$$Y = 0.299R + 0.587G + 0.114B \quad (3.7.1)$$

$$C_b = -0.169R - 0.331G + 0.500B = 0.564(B - Y) \quad (3.7.2)$$

$$C_r = 0.500R - 0.418G - 0.081B = 0.713(R - Y) \quad (3.7.3)$$

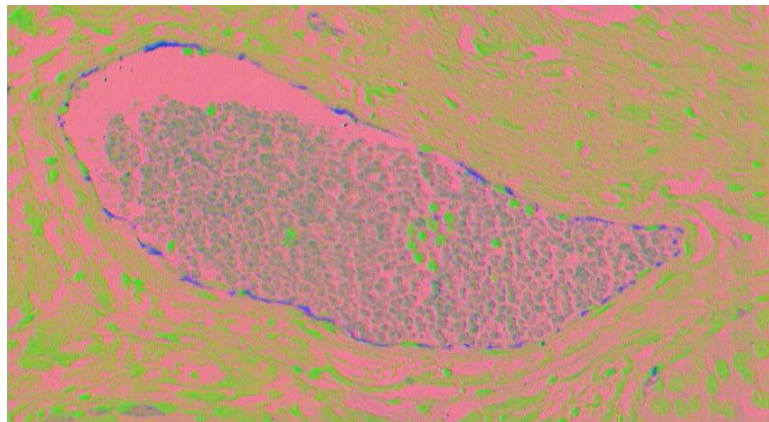


Figure 15: Image showing ROI in  $YC_bC_r$  color space.



### 3.8 Opponent Color Space

*Opponent* color space [2] has been inspired by the physiology of the human visual system (color opponent processes) and therefore is sometimes called physiologically motivated color space. Ewald Hering proposed color opponency theory, which explained some perceptual color phenomenon inexplicable by classical trichromatic theory. Illusions called ‘negative after-images’ and observations during color naming experiments that reddish-green and yellowish-blue colors are not identified, led Hering to suppose that red must be the ‘opposite’ color to green and likewise blue must be ‘opposite’ to yellow. He presumed the existence of three opponent channels (processes) in the human visual system: red-green  $R-G$  channel, yellow-blue  $Ye-B$  channel and achromatic (white-black)  $Wh-Bl$  channel. Herings opponent process theory was one of the first approaches to separate luminance from chrominance [2].

$$RG = R - G \quad (3.8.1)$$

$$YeB = 2B - R - G \quad (3.8.2)$$

$$WhBl = R + G + B \quad (3.8.3)$$

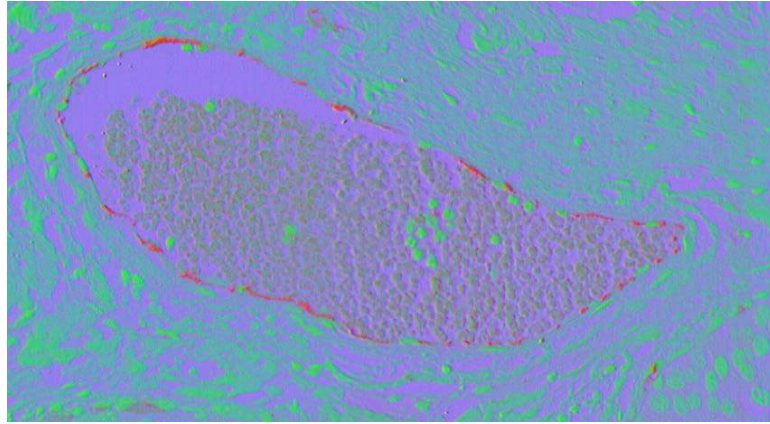


Figure 16: Image showing ROI in *Opponent* color space.

### 3.9 *Ohta* $I_1I_2I_3$ Color Space

The *Ohta* color space [2] was derived as a result of a search of completely statistically independent components on a representative sample of images. The *Ohta* components are a good approximation of the Karhunen-Loeve transformation which is the best in respect of decorrelation of *RGB* components [2].

$$I_1 = \frac{(R+G+B)}{3} \quad (3.9.1)$$

$$I_2 = \frac{(R-B)}{2} \quad (3.9.2)$$

$$I_3 = \frac{(2G-R-B)}{4} \quad (3.9.3)$$

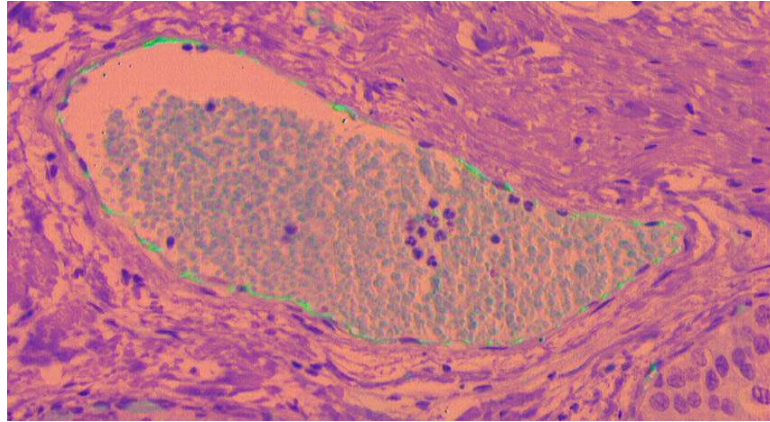


Figure 17: Image showing ROI in *Ohta* color space.

### 3.10 *IHS* Color Space

In the perception of colors, a layperson cannot make intuitive estimates of blue, green and red components, but can easily recognize basic attributes of color: intensity (brightness, lightness)  $I$ , hue  $H$  and saturation  $S$  [2]. The hue  $H$  represents the impression related to the dominant wavelength of the color stimulus. The saturation corresponds to relative color purity (lack of white in the color) and in the case of a pure color it is equal to 100%. (E.g. vivid red  $S = 100\%$  and pale red (pink)  $S = 50\%$ ). Colors with zero saturation are grey levels. Maximum intensity is sensed as pure white, and minimum intensity is as pure black. The simpler formulae for *RGB* to *IHS* color space conversion has been used and is given by.

$$\text{if } \min(R, G, B) = B \text{ then} \quad (3.10.1)$$

$$H = \frac{(G - B)}{3(R + G - 2B)}$$

else if  $\min(R, G, B) = R$  then

$$H = \frac{(B - R)}{3(G + R - 2B)} + \frac{1}{3}$$

else if  $\min(R, G, B) = G$  then

$$H = \frac{(R - G)}{3(R + B - 2G)} + \frac{2}{3}$$

$$S = 1 - 3 \min(r, g, b) = 1 - 3 \frac{\min(R, G, B)}{R + G + B} \quad (3.10.2)$$

$$I = \frac{R + G + B}{3} \quad (3.10.3)$$

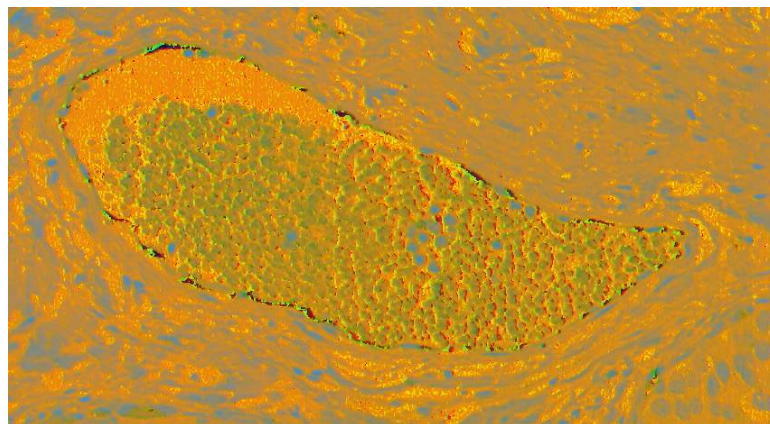


Figure 18: Image showing ROI in *IHS* color space.

### 3.11 *HSV* Color Space

Another model similar to the *IHS* space is the *HSV* space [2]. It differs in formulae for the values of intensity and saturation.

$$\min = \text{minimum}(r, g, b) \quad (3.11.1)$$

$$\max = \text{maximum}(r, g, b) \quad (3.11.2)$$

$$V = \max \quad (3.11.3)$$

$$\text{if } (\max \neq 0) \text{ then} \quad (3.11.4)$$

$$S = \frac{(\max - \min)}{\max}$$

*else*

$$S = 0$$

*end*

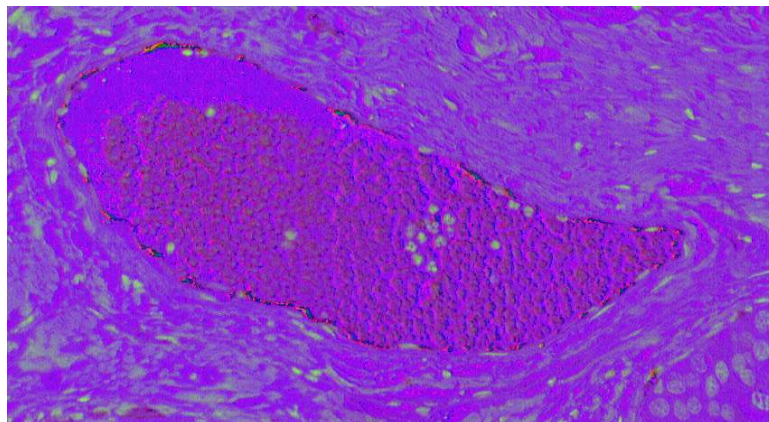


Figure 19: Image showing ROI in *HSV* color space.

### 3.12 *HLS* Color Space

Another model similar to the *IHS* and *HSV* spaces is the *HLS* space [2]. It also differs in formulae for the values of intensity and saturation.

$$\min = \text{minimum}(r, g, b) \quad (3.12.1)$$

$$\max = \text{maximum}(r, g, b) \quad (3.12.2)$$

$$L = \frac{(\max + \min)}{2} \quad (3.12.3)$$

$$\text{if } (\max == \min) \text{ then} \quad (3.12.4)$$

$$S = 0$$

$$\text{else if } (L \leq 0.5) \text{ then}$$

$$S = \frac{(\max - \min)}{(\max + \min)}$$

*else*

$$S = \frac{(\max - \min)}{(2 - \max - \min)}$$

*end*

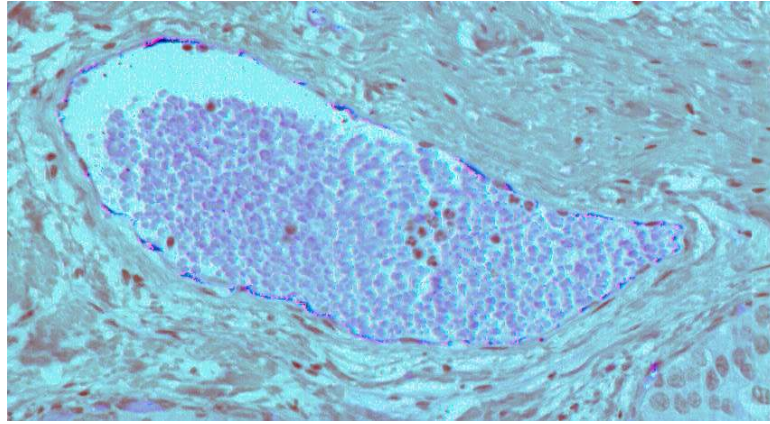


Figure 20: Image showing ROI in *HLS* color space.

### 3.13 *CIELAB* Color Space

The *CIELAB* space [2] is one of the approximately uniform color spaces, where  $L$  stands for lightness value and is orthogonal to  $a$  and  $b$ .  $L$  takes into consideration the non-linear relation between the lightness and the luminance of a point. The component  $a$  denotes relative redness-greenness and  $b$  yellowness-blueness. It is defined by the following expressions, where  $(X_0, Y_0, Z_0)$  represents reference white.

$$L = 116f\left(\frac{Y}{Y_0}\right) - 16 \quad (3.13.1)$$

$$a = 500 \left[ f\left(\frac{X}{X_0}\right) - f\left(\frac{Y}{Y_0}\right) \right] \quad (3.13.2)$$

$$b = 200 \left[ f\left(\frac{Y}{Y_0}\right) - f\left(\frac{Z}{Z_0}\right) \right] \quad (3.13.3)$$

where

$$f(x) = \begin{cases} x^{\frac{1}{3}}, & x > 0.008856 \\ 7.787x + \frac{16}{116}, & \text{otherwise} \end{cases} \quad (3.13.4)$$

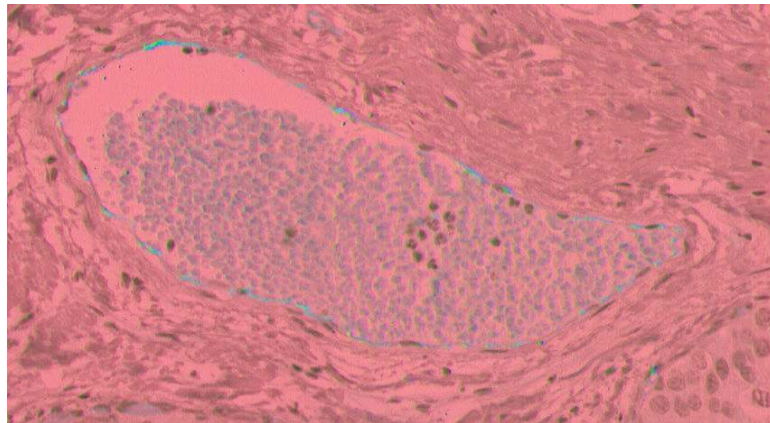


Figure 21: Image showing ROI in *CIELAB* color space.

### 3.14 *CIELUV* Color Space

The *CIELUV* space [2] has component *L* identical to that of the *CIELAB* space, but it should also be noted that the *u* and *v* components are unrelated to the *U* and *V* in *YUV* space. The expressions are given as follows.



$$L = 116f\left(\frac{Y}{Y_0}\right) - 16 \quad (3.14.1)$$

$$u = 13L(u' - u'_0) \quad (3.14.2)$$

$$v = 13L(v' - v'_0) \quad (3.14.3)$$

where

$$f(x) = \begin{cases} x^{\frac{1}{3}}, & x > 0.008856 \\ 7.787x + \frac{16}{116}, & \text{otherwise} \end{cases} \quad (3.14.4)$$

and

$$u' = \frac{4X}{(X + 15Y + 3Z)} \quad (3.14.5)$$

$$v' = \frac{9Y}{(X + 15Y + 3Z)} \quad (3.14.6)$$

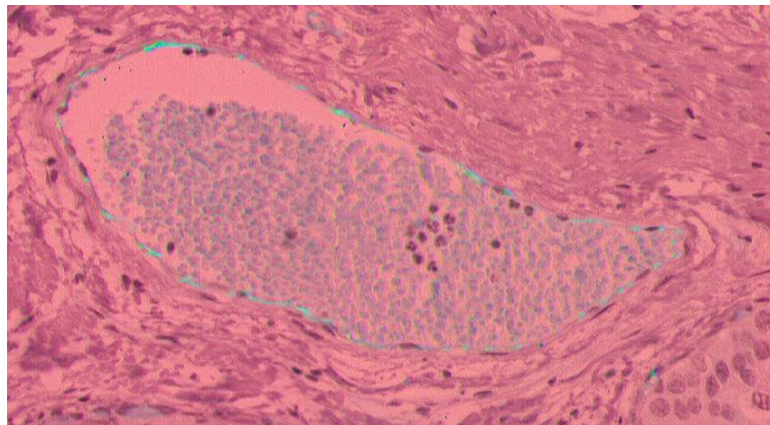


Figure 22: Image showing ROI in *CIELUV* color space.

As seen in all the above images, the color information present in the ROIs are not significantly different from the background (the best case seen in the *Ohta* colorspace), causing the color profiling technique to provide unacceptable segmentations for dataset 1, in all of the colorspace mentioned in this chapter.

## Chapter 4

### 4. IMAGE SEGMENTATION SCHEMES

This chapter explains the broad classification of image segmentation schemes, as well as certain methods pertaining to our project that were implemented and used as benchmarks to compare against.

The segmentation is required in order to aid the oncologist in researching the effects of the tumor on the blood vasculature surrounding it. Therefore, there is a need to provide a fast and reliable algorithm to accurately segment all the ROIs present in a particular high-resolution image.

The existing automated segmentation schemes can be classified mainly into four categories: thresholding techniques, boundary-based techniques, region-based techniques and hybrid techniques [3].

#### 4.1 Thresholding techniques

Thresholding techniques are described by Sahoo et al. in [4] and by Pal et al. in [5]. These techniques are used to obtain good segmentation in images that have regions containing pixels whose intensity values (grayscale, color, etc.) lie within a certain range and are different from the range of values of the background. These techniques neglect the spatial correlation within regions, so the performance is often inadequate.

## 4.2 Boundary-based techniques

Boundary-based techniques are used to segment images where the boundaries of the regions to be segmented have a distinctive change in pixel intensity. The most common method of finding region boundaries is by using edge detectors such as the Sobel operator, to find sharp changes in pixel values. The edge detectors usually provide discontinuous boundaries and may require post-processing such as gap-filling, smoothing, or curve fitting to obtain continuous edge boundaries. Another popular boundary-based segmentation method is the active contour model or snakes algorithm described by Kass et al. [6], as well as the GVF snake described by Xu and Prince [7]. While segmentation is fairly accurate in images that have a well-defined edge or boundary between the ROI and the background, these boundary-based methods are typically computationally intensive.

## 4.3 Region-based techniques

Region-based techniques are particularly effective in images that contain regions with somewhat homogeneous intensity values. Adams and Bischof [8] describe a seeded region growing method where the regions to be segmented are iteratively grown from initial points in the image called seed pixels. Another example of region-based segmentation is the split-and-merge approach described in [9][10]. The region-based techniques require the selection of appropriate seed points and homogeneity criteria in order to produce accurate segmentations.

#### **4.4 Hybrid techniques**

Hybrid techniques involve the integration of more than one of the techniques described above to achieve a final accurate segmentation. Pavlidis et al. [11], Haddon and Boyce [12], Zhu et al. [13], Fan et al. [14] all describe hybrid segmentation methods where the segmentation is performed by unifying region growing schemes with various other schemes. The hybrid techniques are usually data-dependant and involve computationally complex algorithms that are used for the automated segmentation, making them unfeasible for high-resolution image segmentation.

#### **4.5 3-D Histogram method**

Gaddipati et al. [15] have explained a method for automated segmentation of immunohistochemically stained slides. This method involves a transformation into the HSV color space and the creation of a 3-D histogram.

The automated classification system is based on three-dimensional histograms and designed based on the color invariance of stained tissues. Tissue clusters are enclosed in the 3-D histogram interactively using a few representative images. The HSV color space is used to reduce correlation between color features. Depth shaded projections of the 3-D histogram along the three color feature axes are displayed. The projections of the histogram along the three color axes are used to enclose the tissue clusters [15], thereby segmenting the blood vasculature from the background.

The method [15] assumes color invariance in immunohistochemical slides and proceeds to use a fixed 3-D area in the 3-D histogram projections as the foreground regions. The method is useful for images that have separate, well-defined clusters that represent a particular region in the image, either the regions of interest (ROIs) or the background.

Most manual immunohistochemical staining methods results in the antibody (ROI) reacting with the protein to produce a different color as compared to the rest of the background tissue. As a result, different staining runs produce slight differences in stain retention by the ROIs. Consequently, the region clusters in the 3-D histogram are also different for each staining run, making it difficult to segment the ROIs. When the ROIs retain only a small amount of the stain, the clusters overlap each other, thereby making the segmentation process extremely difficult. This is especially true in cancerous-tissue images, where the ROIs (blood vessels, tumors, etc.) appear as multitudes of small circular cells interspersed within background tissue, as seen in Fig. 1.

The original images also have very high intra-slice resolution,  $3200 \times 1600$  pixels, corresponding to 20X magnification. The thickness of the slices limits the inter-slice resolution, so we have to rely on a 2-D approach to segmentation.

#### **4.6 Pulse Coupled Neural Network method (PCNN)**

Stewart et al. [16] explain a method for automated segmentation using region growing with pulse-coupled neural networks. In image processing applications, the

typical network topology for the PCNN is a 2-D planar lattice with a one-to-one pixel to neuron correspondence and for image segmentation, synchronization of a connected set of neurons corresponds to region labeling. In discrete time, connected neurons pulsing at the same time are associated with a unique label while groups pulsing at different times are given distinct labels. However, due to the dynamics of the PCNN, multiple connected groups of neurons will often pulse at the same time, calling for further processing to identify distinct regions. By adopting a novel region growing approach and introducing statistical control, the PCNN can be made to automatically identify a single connected region at each time step without the benefit of predetermined seed locations [16].

PCNN neurons feature feeding  $F$  and linking  $L$  input channels are combined in a multiplicative, excitatory fashion to form an internal activation signal  $U$ , which is compared with a dynamic threshold  $T$  to form output pulses  $Y$ .

In the linking field model (LFM)  $F$ ,  $L$  and  $T$  are all instantiated as leaky capacitors. The feeding input at time  $t$  is reduced to the corresponding pixel intensity  $F_x[t] = G_x$  and linking input  $L_x$  is given by

$$L_x[t] = \sum_{z \in N(x)} Y_z[t] - d \quad (4.6.1)$$

where the first term sums over the output of the nearest 8 neighbors and the second term  $d$  is a positive constant which is the preferred method of inhibition. Along with other modifications, the simplified feeding and linking inputs arguably compromise biological plausibility in search of improved efficiency and segmentation performance.

The change introduced by this method is *temporal independence* where each time step of the PCNN is treated as an independent process. The excitatory signal is given by

$$U_x[t] = G_x \{1 + \beta_t L_x[t]\} \quad (4.6.2)$$

Output values are given by a simple step function comparing  $U_x$  with a dynamic threshold  $T_x$

$$Y_x[t] = \begin{cases} 1, & U_x[t] > T_x[t] \\ 0, & \text{otherwise} \end{cases} \quad (4.6.3)$$

Since the equations are visited in sequence, at the beginning of each time step there will be no linking input, and the initial pulsing activity is given by

$$Y_x[t] = \begin{cases} 1, & G_x > T_x[t] \\ 0, & \text{otherwise} \end{cases} \quad (4.6.4)$$

Eqs. (4.6.1) – (4.6.3) are iterated until the output  $Y$  converges to a fixed state with convergence guaranteed because the value of  $U_x$  never falls during a time step while  $T_x[t]$  is constant.

Algorithmic termination is resolved by applying the single-pass stopping condition where each neuron in the network is allowed to pulse: change state from zero to one, once and only once [16]. Time of pulsing is used as a region label and these times are stored in the matrix  $P$  with initial values of zero at all locations.



$$P_x[t] = \begin{cases} t, Y_x[t] = 1 \\ P_x[t-1], otherwise \end{cases} \quad (4.6.5)$$

By temporal independence threshold decay is abandoned in favor of explicit working thresholds  $wt_l$  for all neurons yet to pulse. These neurons and the corresponding pixels are referred to as *active*. The threshold for other neurons is set to a large constant value  $\Omega$  to prevent repeated pulsing at the same location.

$$T_x[t] = \begin{cases} wt_l, P_x[t-1] = 0 \\ \Omega, otherwise \end{cases} \quad (4.6.6)$$

The algorithm is as follows

1. Set  $d$  and  $\Omega$
2. Repeat
  - a.  $wt_l = \max[G_y] \forall y | P_y[t-1] = 0$
  - b.  $T_x[t] = \begin{cases} wt_l, P_x[t-1] = 0 \\ \Omega, otherwise \end{cases}$
  - c.  $Y_x[t] = \begin{cases} 1, G_x > T_x[t] \\ 0, otherwise \end{cases}$
3. Set  $\beta_l$  sufficient for initial spread
4. *While* statistical termination condition not met
  - a. *While* there is any change in pulsing activity

- i.  $L_x[t] = \sum_{x \in N(x)} Y_z[t] - d$

$$\text{ii. } U_x[t] = G_x \{1 + \beta_t L_x[t]\}$$

$$\text{iii. } Y_x[t] = \begin{cases} 1, & U_x[t] > T_x[t] \\ 0, & \text{otherwise} \end{cases}$$

*End*

$$\beta_t = \beta_t + \delta\beta$$

*End*

$$5. P_x[t] = \begin{cases} t, & Y_x[t] = 1 \\ P_x[t-1], & \text{otherwise} \end{cases}$$

$$6. t = t + 1$$

The time or iteration of pulsing is used as a region label; all neurons pulsing at a particular iteration are considered to be one region type, and all neurons pulsing at a different iteration are considered to be part of a different region type [16].

#### **4.7 k-means Clustering method**

The k-means clustering method described in [17] and [18] segments the image into  $n$  clusters depending on the Mahalanobis distance of each pixel to the centroid of the cluster. The number of clusters  $n$  is chosen depending upon the expected number of clusters corresponding to different biological components and consequently different color retention in the images. An image that has been segmented into 12 clusters by using

the k-means algorithm [18] is shown in Fig. 23. A cursory observation of the segmentation shows the different clusters, with a major portion of our ROIs corresponding to a particular cluster, and small regions inside the ROIs corresponding to a cluster that also includes a number of background pixels.

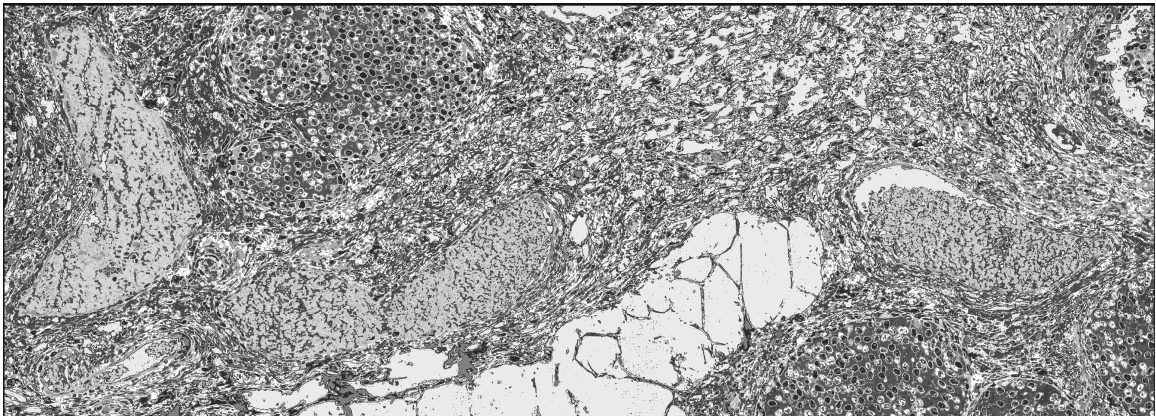


Figure 23: A k-means cluster showing 12 clusters.

The clustered image is then thresholded to extract the cluster that corresponds to the ROIs, as shown in Fig. 24. The failure of k-means to accurately classify the ROIs into one or more clusters necessitates the use of a region-based segmentation scheme to accurately segment the ROIs from the background.

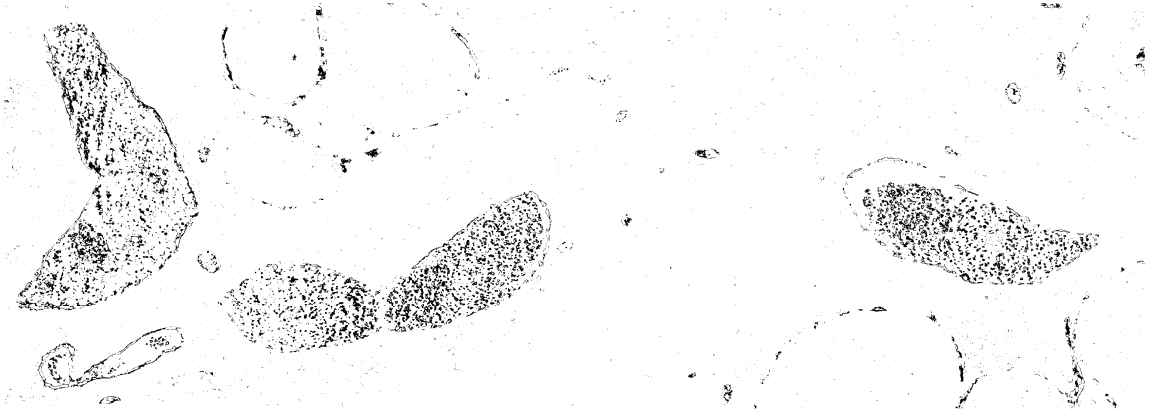


Figure 24: A thresholded k-means cluster segmentation.

#### 4.8 Integrated Color Edges method

Fan et al. [14] explain a method for automatic image segmentation that integrates color edges and seeded region growing [8]. The image is transformed into YUV (luma and color difference) color space, and the pixels are classified into two opposite classes, edge and non-edge, based on the local maximum edge strength of the pixel being greater (or lesser, respectively) than a threshold determined by the fast entropic thresholding technique. The union of the set of Y, U and V edge pixels forms the color edges [14]. The obtained color edges are useful for providing the candidate seed pixels of the image. The number of seed pixels produced by this method is still high on average, thereby making it computationally intensive for high-resolution images.

## 4.9 Histogram Peak Selection method

Silva et al. [19] explain a method for seed pixel selection, wherein the seed pixels and the homogeneity criteria for the region growing algorithm [8] are selected based on the image histogram. The main contribution of this scheme is the method of seed pixels selection.

The method involves the implementation of a new peak detection algorithm based on the image histogram. This algorithm provides the seed pixels and the homogeneity criteria to the region growing algorithm.

The algorithm is as follows

1. Build the image histogram.
2. Determine all the peaks ( $P$ ) of the histogram.
3. Select, in sequence, some or all the peaks  $P$  between the ones that were determined in step. 2, following the rule
  - a. For each  $P$  calculate  $g(p_i)$

$$g(p_i) = freq_i \times dist_i^2 \quad (4.9.1)$$

where  $p_i$  is the peak  $i$  of the histogram,  $freq_i$  is the height of the peak and  $dist_i = 1$  if  $p_i$  is the first peak, or the horizontal distance to the nearest selected  $p_i$ , otherwise

- b. The peak  $p_i$  which maximizes the function  $g(p_i)$  is selected.
4. All the pixels in the image with gray level of the selected peaks  $p_i$  from  $P$  are chosen as seed pixels.

If  $f(x,y)$  is the gray level of the pixel  $(x,y)$ , the homogeneity criteria ( $H_{p_i}$ ) is defined as

$$H_{p_i}(x,y) = \begin{cases} True, & v_i \leq f(x,y) \leq v_{i+1} \\ False, & otherwise \end{cases} \quad (4.9.2)$$

where the peak  $p_i$  being tested is located between one valley on the left  $v_i$  and one valley on the right  $v_{i+1}$ . The computation of the gray level of the valleys is presented in Eq. (4.9.3) below.

For each peak  $p_i$  select a gray level  $I$  between  $p_i$  and  $p_{i+1}$  that minimize the function  $h_i(I)$  as defined in Eq. (4.9.3) below.

$$h_i(I) = freq_i \times \left| I - \frac{(I_i + I_{i+1})}{2} \right| \quad (4.9.3)$$

$$v_i = \underset{i}{Min}(h_i(I)) \quad (4.9.4)$$

where  $I$  is a gray level between the gray levels of two selected peaks  $p_i$  and  $p_{i+1}$ ;  $freq_i$  is the frequency of  $I$ ;  $\frac{(I_i + I_{i+1})}{2}$  is the middle value between  $p_i$  and  $p_{i+1}$ , and;

$\left| I - \frac{(I_i + I_{i+1})}{2} \right|$  is the distance of  $I$  to the middle point between  $p_i$  and  $p_{i+1}$ .

The peaks of the histogram are determined [19], and an appropriate number of these peaks are selected based on the frequency of pixels that have the intensity represented by the peak, and based on the distance of the peak from other previously selected peaks. All the peaks that satisfy the given conditions are selected, and all the pixels whose intensity equals the intensities represented by these peaks are chosen as candidate seed pixels [19].

The number of seed pixels produced by all selected peaks [19] is extremely high, making it unfeasible for high-resolution images. Restricting the number of candidate seed pixels, by restricting the number of histogram peaks chosen, results in inaccurate segmentations. Thus, an alternative method for accurate selection of seed pixels, resulting in accurate segmentations, is needed.

## Chapter 5

### 5. PRINCIPAL COMPONENT ANALYSIS

This chapter explains principal component analysis, which is used as an initial step to convert the 3-D color images to a 1-D grayscale intensity image.

Principal component analysis is mathematically defined [20] as an orthogonal linear transformation that transforms the data to a new coordinate system such that the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinate, and so on.

The principal components of the RGB color images are obtained using the covariance method [20] explained below.

One approach to coping with the problem of excessive dimensionality is to reduce the dimensionality by combining features. Linear combinations are particularly attractive because they are simple to compute and analytically tractable. In effect, linear methods project the high-dimensional data onto a lower dimensional space. The principal component analysis approach seeks a projection that best represents the data in a least-squares sense.

The problem consists of representing all of the vectors in a set of  $n$  d-dimensional samples  $x_1, \dots, x_n$  by a single vector  $x_0$  [20]. To be more specific, suppose that we want to



find a vector  $x_0$  such that the sum of the squared distances between  $x_0$  and the various  $x_k$  is as small as possible. The squared-error criterion function  $J_0(x_0)$  is defined by

$$J_0(x_0) = \sum_{k=1}^n \|x_0 - x_k\|^2 \quad (5.1)$$

and seek the value of  $x_0$  that minimizes  $J_0$ . It is simple to show that the solution to this problem is given by  $x_0 = m$ , where  $m$  is the sample mean

$$m = \frac{1}{n} \sum_{k=1}^n x_k \quad (5.2)$$

This can be easily verified by writing

$$\begin{aligned} J_0(x_0) &= \sum_{k=1}^n \|(x_0 - m) - (x_k - m)\|^2 \\ &= \sum_{k=1}^n \|x_0 - m\|^2 - 2 \sum_{k=1}^n (x_0 - m)^t (x_k - m) + \sum_{k=1}^n \|x_k - m\|^2 \\ &= \sum_{k=1}^n \|x_0 - m\|^2 - 2(x_0 - m)^t \sum_{k=1}^n (x_k - m) + \sum_{k=1}^n \|x_k - m\|^2 \\ &= \sum_{k=1}^n \|x_0 - m\|^2 + \sum_{k=1}^n \|x_k - m\|^2 \end{aligned} \quad (5.3)$$

where the second sum is independent of  $x_0$ . This expression is obviously minimized by the choice of  $x_0 = m$ .

The sample mean is a zero-dimensional representation of the data set. It is simple, but it does not reveal any of the variability in the data. We can obtain a more interesting, one-dimensional representation by projecting the data onto a line running through the sample mean [20]. Let  $e$  be a unit vector in the direction of the line. Then the equation of the line can be written as

$$x = m + \alpha e \quad (5.4)$$

where the scalar  $\alpha$  (which takes on any real value) corresponds to the distance of any point  $x$  from the mean  $m$ . If we represent  $x_k$  by  $m + \alpha_k e$  we can find an “optimal” set of coefficients  $\alpha_k$  by minimizing the squared-error criterion function.

$$\begin{aligned} J_1(\alpha_1, \dots, \alpha_n, e) &= \sum_{k=1}^n \| (m + \alpha_k e) - x_k \|^2 \\ &= \sum_{k=1}^n \| \alpha_k e - (x_k - m) \|^2 \\ &= \sum_{k=1}^n \alpha_k^2 \|e\|^2 - 2 \sum_{k=1}^n \alpha_k e^t (x_k - m) + \sum_{k=1}^n \|x_k - m\|^2 \end{aligned} \quad (5.5)$$

recognizing that  $\|e\| = 1$ , partially differentiating with respect to  $\alpha_k$  and setting the derivative to zero, we obtain

$$\alpha_k = e^t (x_k - m) \quad (5.6)$$

Geometrically, this result merely says that we obtain a least-squares solution by projecting the vector  $x_k$  onto the line in the direction of  $e$  that passes through the sample mean [20].

This brings us to the more interesting problem of finding the *best* direction  $e$  for the line. The solution to this problem involves the so-called *scatter matrix*  $S$  defined by

$$S = \sum_{k=1}^n (x_k - m)(x_k - m)^t \quad (5.7)$$

The scatter matrix should look familiar – it is merely  $n - 1$  times the sample covariance matrix. It arises here when we substitute  $\alpha_k$  found in Eq. (5.6) into Eq. (5.5) to obtain

$$\begin{aligned} J_1(e) &= \sum_{k=1}^n \alpha_k^2 - 2 \sum_{k=1}^n \alpha_k^2 + \sum_{k=1}^n \|x_k - m\|^2 \\ &= -\sum_{k=1}^n [e^t(x_k - m)]^2 + \sum_{k=1}^n \|x_k - m\|^2 \\ &= -\sum_{k=1}^n e^t(x_k - m)(x_k - m)^t e + \sum_{k=1}^n \|x_k - m\|^2 \\ &= -e^t S e + \sum_{k=1}^n \|x_k - m\|^2 \end{aligned} \quad (5.8)$$

Clearly, the vector  $e$  that minimizes  $J_I$  also maximizes  $e'Se$ . The method of Lagrange multipliers described in [20] is used to maximize  $e'Se$  subject to the constraint that  $\|e\|=1$ . Letting  $\lambda$  be the undetermined multiplier, we differentiate

$$u = e'Se - \lambda(e'e - 1) \quad (5.9)$$

with respect to  $e$  to obtain

$$\frac{\partial u}{\partial e} = 2Se - 2\lambda e \quad (5.10)$$

Setting this gradient vector equal to zero, we see that  $e$  must be an eigenvector of the scatter matrix

$$Se = \lambda e \quad (5.11)$$

In particular because  $e'Se = \lambda e'e = \lambda$ , it follows that to maximize  $e'Se$ , we want to select the eigenvector corresponding to the largest eigenvalue of the scatter matrix. In other words, to find the best one-dimensional projection of the data (best in the least-sum-of-squared-error sense), we project the data onto a line through the sample mean in the direction of the eigenvector of the scatter matrix having the largest eigenvalue [20].

This result can be readily extended from a one-dimensional projection to a  $d'$ -dimensional projection. In place of Eq. (5.4) we write,

$$x = m + \sum_{i=1}^{d'} \alpha_i e_i \quad (5.12)$$

where  $d' \leq d$ . It is not difficult to show that the criterion function

$$J_{d'} = \sum_{k=1}^n \left\| \left( m + \sum_{i=1}^{d'} \alpha_{ki} e_i \right) - x_k \right\|^2 \quad (5.13)$$

is minimized when the vectors  $e_1, \dots, e_{d'}$  are the  $d'$  eigenvectors of the scatter matrix having the largest eigenvalues. Because the scatter matrix is real and symmetric, these eigenvectors are orthogonal. They form a natural set of basis vectors for representing any feature vector  $x$ . The coefficients  $\alpha_i$  in Eq. (5.12) are the components of  $x$  in that basis, and are called the *principal components*. Geometrically, if we picture the data points  $x_1, \dots, x_n$  as forming a  $d$ -dimensional, hyperellipsoidally shaped cloud, then the eigenvectors of the scatter matrix are the principal axes of that hyperellipsoid [20]. Principal component analysis reduces the dimensionality of feature space by restricting attention to those directions along which the scatter of the cloud is greatest [20].

In our case, the RGB color image has three dimensions and the features are combined to form a reduced dimension grayscale image. Thus principal component analysis is used to reduce the dimensionality of the data from a color image into a grayscale image. The RGB images are transformed into three grayscale images, each representing the three principal components in the original image. Deveaux et al. [21] explain that the major advantage of using principal component analysis is that, theoretically the information on

the three axes is completely decorrelated [21], and therefore the transformed images can be independently and separately processed. Another major factor that influences the use of the principal component analysis is that, theoretically the luminance information is concentrated on the  $\lambda_1$  axis, and the chrominance information is contained on the  $\lambda_2$  and  $\lambda_3$  axes [21]. The major information distinguishing the ROIs from the background is the chrominance information, and this is retained in the second principal component.

The second advantage of the principal component analysis is the conservation of energy through coordinate transformation. The total energy contained in the transformed image is the sum of the individual energies of each principal component image ( $Energy(K) = \lambda_1 + \lambda_2 + \lambda_3 = 100\%$ ), and the energy of each axis is given by its associated eigenvalue [21]. Since the transformed data is arranged in descending order of eigenvalues, and therefore by definition  $\lambda_1 > \lambda_2 > \lambda_3$  the  $\lambda_1$  axis contains more energy than the  $\lambda_2$  axis, and the  $\lambda_3$  axis is weak in energy. To measure the importance of energy on the  $\lambda_3$  axis, the representation quality factor  $Q$  [21] is calculated as

$$Q = \frac{\lambda_1 + \lambda_2}{\lambda_1 + \lambda_2 + \lambda_3} \quad (5.14)$$

$Q$  measures the fraction of energy preserved when the  $\lambda_3$  axis is removed. In this application, we need to perform image segmentation primarily on the chrominance information, so the  $\lambda_1$  and  $\lambda_3$  axes can be removed without significant loss of information. Therefore, the RGB images are transformed into the grayscale  $\lambda_2$  image, which represents the significant chrominance information.

The resulting second principal component image is thresholded (Section 9.5.1), and these thresholded pixels are selected as the candidate seed pixels for region growing.

The principal component analysis threshold yields about 237220 seed pixels for each image, on an average over the entire dataset. Growing the regions using these 237220 seed pixels results in two problems: (a) some of these pixels (especially ROI border pixels) may result in an inaccurate segmentation of the ROIs, with extra background surrounding the ROI being classified as a part of it, and (b) some of the seed pixels may even grow into a region identical to one grown from a previous seed pixel, resulting in a reduction in efficiency and an unnecessary increase in computation time.

Therefore, there is a need to prune the seed pixels down to a more feasible number, eliminating seed pixels yielding inaccurate segmentation, and minimizing the number of seed pixels growing into identical regions. This is achieved by the seed pruning algorithm discussed in the following chapter.

## Chapter 6

### 6. SEED PRUNING ALGORITHM

For each second-principal-component image, we use thresholding (Section 9.5.1) to locate candidate seed pixels, and then prune those seed pixels before performing region growing. The proposed seed-pruning algorithm involves the creation of a Gaussian multi-resolution pyramid [22] by local averaging, and down sampling during which the resolution is halved at each level. We define three categories of seed pixels for a given multi-resolution pyramid:

(a) *Candidate seed pixels* – the set of all pixels that fall within the threshold (Section 9.5.1), at each resolution level, starting at the lowest resolution.

(b) *Potential seed pixels* – the set of all pixels, at each resolution level greater than one, obtained by extrapolating the locations (row and column are multiplied by two) of the *fixed seed pixels* at the next lower resolution level.

(c) *Fixed seed pixels* – the set of all *potential seed pixels* that survive the seed pruning algorithm at each level greater than one and the set of all *candidate seed pixels* at level one.

(d) *Final seed pixels* – the set of all *fixed seed pixels* that survive the seed pruning algorithm at the highest resolution level.



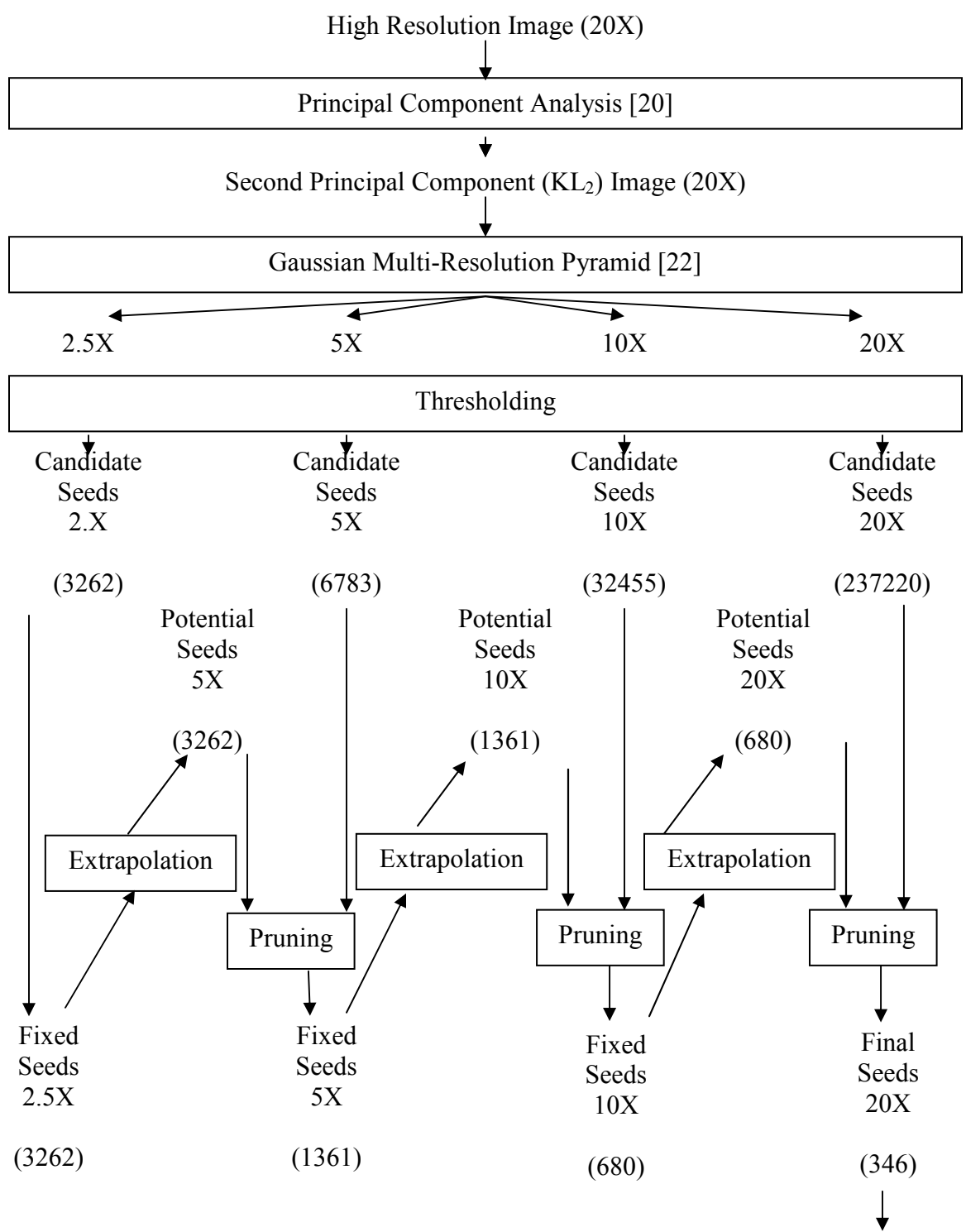


Figure 25: Flowchart showing the working of the Seed Pruning Algorithm with number of each class of pixels for a specific case highlighted at each step.

Considering the specific case of the original image imaged at 20X resolution, we create lower-resolution images at 10X, 5X, and 2.5X to form a four-level pyramid. The major ROIs are still prominently visible at 2.5X, whereas noisy pixels are not so prominent. The number of *candidate seed pixels* at this level is around 3600, which if extrapolated directly to 20X is still large.

The seed pruning algorithm concentrates on minimizing this number as we move through the multi-resolution pyramid. It is analyzed in the following discrete steps.

(1) Create an *m-level* multi-resolution pyramid [22] from the second principal component of the original image, where each image in the pyramid has half the resolution of the image at the previous level.

(2) Threshold (Section 9.5.1) each image in the pyramid and select the set of all pixels falling within the threshold as the set of *candidate seed pixels* at that level.

(3) At the lowest resolution level (level 1), set the *fixed seed pixels* set to be equal to the set of *candidate seed pixels* at that level.

(4) The locations of all *fixed seed pixels* at level  $k$  ( $k=1,2,\dots,m$ ) are extrapolated (row and column are multiplied by two) to yield *potential seed pixels* at level  $k+1$ .

(5) For each *potential seed pixel* at level  $k$  ( $k=2,3,\dots,m$ ) count the number of *candidate seed pixels* in its  $(2k-1) \times (2k-1)$  neighborhood.

(6) If the count size is greater than 50% (Section 9.5.2) of the window size, then the *potential seed pixel* becomes a *fixed seed pixel* at level  $k$ , else it is discarded.

(7) Repeat steps 4 through 6 until the *fixed seed pixels* at the level  $m$  (the full resolution) are obtained.

To determine whether a *potential seed pixel* becomes a *fixed seed pixel*, we examine the number of *candidate seed pixels* appearing in a local neighborhood (step 5). The number of *candidate seed pixels* in this window will be high for *potential seed pixels* that are in the middle of an ROI, and will be low for the *potential seed pixels* that are on the borders of the ROIs. By requiring that at least 50% (Section 9.5.2) of pixels in the window are *candidate seed pixels*, we can eliminate ROI border seed pixels, which often result in inaccurate segmentations.

After this pruning step, we obtain the *fixed seed pixels* at resolution level  $k$ , which are called *final seed pixels* which are much fewer than the number of *candidate seed pixels* at this level, and also lesser than the number of *fixed seed pixels* at level  $k-1$ .

Next, the *fixed seed pixels* at level  $k$  are extrapolated to obtain *potential seed pixel* locations at level  $k+1$ . The neighborhood window size,  $(2k-1) \times (2k-1)$ , increases as  $k$  increases, which is appropriate due to the increase in resolution. The larger window size is necessary for further elimination of any surviving border pixels. The process is repeated until we reach the full-resolution level of the pyramid. Each time, the condition for validating a *potential seed pixel* to a *fixed seed pixel* is the presence of more than 50% (Section 9.5.2) *candidate seed pixels* in its neighborhood.

## Chapter 7

### 7. REGION GROWING

The region growing scheme implemented as the final step of the segmentation process is described by Adams and Bischof [8]. The algorithm is used for the segmentation of intensity images and is robust, rapid and free of tuning parameters. However it requires the input of a number of seeds, either individual pixels or regions, which will control the formation of regions into which the image will be segmented.

Region based methods rely on the postulate that neighboring pixels within a region have a similar value. The procedure is to compare one pixel to its neighbor(s). If a criterion of homogeneity is satisfied, the pixel is said to belong to the same class as one or more of its neighbors.

Seeded region growing [8] performs a segmentation of an image with respect to a set of points, known as seeds. We start with a number of seeds which have been grouped into  $n$  sets, say,  $A_1, A_2, \dots, A_n$ . sometimes, individual sets will consist of single points. It is in the choice of seeds that the decision of what is a feature of interest and what is irrelevant or noise is embedded. This is the most important step of the region growing process, and the seed pruning algorithm in Chapter 6., is responsible for selecting the seed pixels that provide accurate segmentations.

Given the seeds, the algorithm then finds a tessellation of the image into regions with the property that each connected component of a region meets (non-empty intersection with) exactly one of the  $A_i$  and, subject to this constraint the regions are chosen to be as homogenous as possible. Though the method described is applicable to grayscale images, the extension to color or multispectral images is straightforward and involves the choice of a suitable metric in color or multispectral space, or a suitable conversion into the grayscale space. This is accomplished using the principal component analysis mentioned in Chapter 5.

The process evolves inductively from the seeds, namely the initial state of the sets,  $A_1, A_2, \dots, A_n$ . Each step of the algorithm involves the addition of one pixel to one of the above sets. The state of the sets  $A_i$  after  $m$  steps is considered. The set  $T$  of all as-yet unallocated pixels which border at least one of the regions is given by

$$T = \left\{ x \notin \bigcup_{i=1}^n A_i \mid N(x) \cap \bigcup_{i=1}^n A_i \neq \emptyset \right\} \quad (7.1)$$

where  $N(x)$  is the set of immediate neighbors of the pixel  $x$ . The neighbors of the pixel are defined by a rectangular grid with immediate neighbors being those which are 8-connected to the pixel  $x$ .

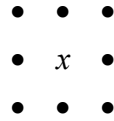


Figure 26: 8-connected neighbor pixels of pixel  $x$ .

If, for  $x \in T$  we have that  $N(x)$  meets just one of the  $A_i$  then  $i(x) \in \{1, 2, \dots, n\}$  is defined to be that index such that  $N(x) \cap A_{i(x)} \neq \emptyset$  and  $\delta(x)$  is defined to be a measure of how different  $x$  is from the region it adjoins and is given by

$$\delta(x) = |g(x) - \underset{y \in A_{i(x)}}{\text{mean}}[g(y)]| \quad (7.2)$$

where  $g(x)$  is the gray value of the image point  $x$ . If  $N(x)$  meets two or more of the  $A_i$ , we take  $i(x)$  to be a value of  $i$  such that  $N(x)$  meets  $A_i$  and  $\delta(x)$  is minimized. Alternatively, we can classify  $x$  as a boundary pixel and append it to the set  $B$  of already found boundary pixels. Flagging such boundary pixels is useful for display purposes or for use with a semi-interactive corrective procedure if needed. A  $z \in T$  is then taken such that,

$$\delta(z) = \min_{x \in T} \{\delta(x)\} \quad (7.3)$$

and the  $z$  is appended to  $A_{i(z)}$ . This completes step  $m + 1$ . The process [8] is repeated until all pixels have been allocated. The process commences with each  $A_i$  being just one of the seed sets. The definitions in (8.2) and (8.3) ensure that the final segmentation is into regions as homogenous as possible given the connectivity constraint [8].

While programming the region growing algorithm [8] a sequentially sorted list (SSL) is used. The SSL is a linked list of objects, in this case pixel addresses, which are ordered according to some attribute. When considering a new pixel, for example, at the beginning of each step, the one at the beginning of the list is taken. When adding a pixel to the list, it is placed according to its value of the ordering attribute, in this case according to  $\delta$ .

The algorithm for implementing seeded region growing [8] is given below

- 1) Label seed points according to their initial grouping
- 2) Put neighbors of seed points (the initial  $T$ ) into the SSL.
- 3) *While* the SSL is *not empty*
  - i. Remove first point  $y$  from SSL
  - ii. Test the neighbors of this point
    1. *If* all neighbors of  $y$  which are already labeled (other than with boundary label) have the same label
      - a. *Set*  $y$  to this label
      - b. *Update* running mean of corresponding region
      - c. *Add* neighbors of neighbors of  $y$  which are neither already set nor already in the SSL to the SSL according to their value of  $\delta$ .
    2. *Else*
      - a. *Flag*  $y$  with the boundary label

In executing the algorithm each pixel is visited just once, although at each visit each of the neighbors are viewed and tested. This makes for a very rapid program.

The *final seed pixels* obtained by the seed pruning algorithm in Chapter 6., are used as the inputs to the region growing scheme [8], along with the grayscale image which is the second principal component image of the original high resolution color image.

Each of these *final seed pixel* locations is used as a starting point  $A_i$  in the principal component image to create an outward iterative growth of a region, which is expected to be a ROI. The seeded region growing algorithm explained above is then implemented to generate the final segmented region ROIs.

The homogeneity criterion used in this method is the second principal component intensity of the pixel under test. If this value falls within a threshold (Section 9.5.3), the pixel under test is classified as a region pixel.

The goals of the seed pruning algorithm are to provide the fewest number of *final seed pixels* that will result in the growth of a number of segmented regions that correspond exactly to the number of ROIs present in the image, and to select the seed pixels that only produce accurately segmented regions, in the seeded region growing step.

Repeating the process for all the *final seed pixels* obtained by the seed pruning algorithm, results in the formation of a corresponding number of grown regions that ideally should represent the ROIs. A check is also performed within the region growing scheme to determine whether the seed pixel currently being tested is part of a region



already obtained from an earlier seed pixel. This minimizes the unnecessary duplication of identical region obtained from multiple seed pixels.

Due to the inherent complexities in the manual segmentation (namely small non-ROI hole pixels appearing as ROIs due to inabilities to accurately manually segment the images, a post processing step that involves morphological closing operation [3] with a  $11 \times 11$  window (Section 9.5.4) to smooth the holes present in the ROIs, is used before accuracy and false negative and false positive ratio calculations.

## Chapter 8

### 8. RESULTS – SEGMENTATED IMAGES



Figure 27: Automated segmentation using our approach.



Figure 28: Manual segmentation.



Figure 29: Automated segmentation using our approach.

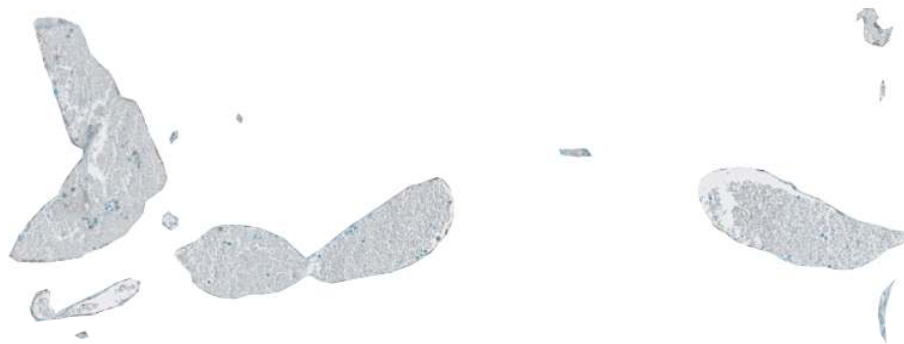


Figure 30: Manual segmentation.



Figure 31: Automated segmentation using our approach.



Figure 32: Manual segmentation.



Figure 33: Automated segmentation using our approach.



Figure 34: Manual segmentation.



Figure 35: Automated segmentation using our approach.



Figure 36: Manual segmentation.



Figure 37: Automated segmentation using our approach.



Figure 38: Manual segmentation.

## Chapter 9

### 9. RESULTS – ANALYSIS AND DISCUSSION

The original images are high-resolution ( $3200 \times 1600$  pixels, corresponding to 20X magnification) 24 bits per pixel digital color images of histological slides digitized using a DMetrix scanner. The histological slides contain slices of breast cancer tissue that has been stained with a coloring agent (Cd-31 coloring stain). The regions of interest (tumors, blood vessels, etc.) retain the stain, and have a difference in color with respect to the background tissue.

#### 9.1 Segmentation Accuracy

Segmentation accuracy is based on the area of mutual overlap [3]. If  $P_1$  is the number of segmented region pixels obtained using our approach, and  $P_2$  is the number of region pixels in the manually segmented regions, and  $MO$  is the mutual overlap between automatically segmented regions and manually segmented regions, then the performance metric for accuracy [3] is given by

$$acc = \frac{2MO}{P_1 + P_2} \quad (9.1)$$

The mean area of mutual overlap over a sample set of eight images each containing about twenty-four ROIs was found to be 85.2% with a standard deviation of 5.7 percentage points.



## 9.2 Efficiency in reducing number of seed pixels

We also measured the efficiency of the algorithm in terms of percentage reduction in the number of seed pixels as compared with (a) the integrated color edge method [14], (b) using the histogram method [19] for finding the seed pixels, (c) using all the candidate seed pixels present in the lowest-resolution image extrapolated to the original resolution level (multiplying row and column by  $2^{m-1}$ ), and (d) using all of the candidate seed pixels of the highest-resolution image (too many seeds to perform region growing) and (e) the k-means clustering method [18] (too many seeds to perform region growing). The accuracy of our seed pruning algorithm was also compared with these techniques, and also the (f) pulse-coupled neural network method [16] (the pulse-coupled neural network method does not generate seeds and does not use region growing).

Table 1: NUMBER OF SEED PIXELS AND SEGMENTATION ACCURACY

Method	No. of Seeds	Accuracy (%)
Seed pruning	602	85.2
Color edges [14]	883	73.9
Histogram peaks [19]	1102	80.6
k-means Clustering [18]	122303	-
Pulse-coupled NN [16]	-	45.9

Table 1. shows the number of seed pixels generated by our seed pruning algorithm compared with various existing algorithms and their corresponding segmentation accuracies.

Our proposed seed pruning algorithm achieves a 31% reduction in number of seed pixels selected, compared to the integrated color edge method [14], a 45% reduction compared to the histogram peaks method [19], and a 99% reduction in candidate seed pixels compared to the k-means clustering method [18]. The accuracy is also higher than all the other methods that could be feasibly implemented.

### 9.3 Effect of number of resolution levels on efficiency and accuracy

We measured the effect of the seed pruning algorithm in pruning the seeds at different resolutions levels in the multi-resolution pyramid, and computed the accuracy of the segmentation each time. The results are shown in Table. 2.

Table 2: NUMBER OF SEED PIXELS AND SEGMENTATION ACCURACY OF THE SEED PRUNING METHOD BY VARYING RESOLUTION LEVELS

Seed Pruning Resolution	No. of Seeds	Accuracy (%)
1.250X	73	23.7
2.500X	602	85.2
5X	4232	85.2
10X	25512	-
20X	237220	-

The number of resolution levels that the seed pruning algorithm uses to prune the number of seed pixels depends on various image factors, like initial resolution level, image size etc.

By using lower resolution levels, the number of seed pixels selected by the seed pruning algorithm reduces each time as expected. But, below the fourth resolution level the seed pixels selected produce inaccurate segmentations. This is largely due to smaller ROIs being completely invisible at 1.25X. Thus 2.5X has been selected as the lowest resolution level, for the seed pruning algorithm to select seed pixels for region growing segmentation.

#### **9.4 False Negative Ratio and False Positive Ratio**

We also measured the accuracy of segmentation in terms of the false negative ratio (FNR) and false positive ratio (FPR) [23]. The false positive error [23] is defined as the number of non ROI pixels in the manually segmented image that are misclassified as ROI pixels in the automated segmentation. The false positive ratio (FPR) [23] is then computed relative to the number of ROI pixels found in the manually segmented data.

$$\text{False\_Positive\_Ratio}(FPR) = \frac{\text{False\_Positive\_Pixels}}{\text{Total\_ROI\_Pixels}} \quad (9.2)$$

The false negative error [23] is defined as the number of ROI pixels in the manually segmented image that are misclassified as non ROI pixels in the automated segmentation.

The false negative ratio (FNR) [23] is then computed relative to the number of ROI pixels found in the manually segmented data.

$$\text{False\_Negative\_Ratio}(FPR) = \frac{\text{False\_Negative\_Pixels}}{\text{Total\_ROI\_Pixels}} \quad (9.3)$$

Table 3: FALSE NEGATIVE AND FALSE POSITIVE RATIOS

Method	FNR (%)	FPR (%)
Seed pruning	15.94	12.74
Color edges [14]	26.11	21.06
Histogram peaks [19]	22.72	11.53

Table 3. shows the false positive and false negative ratios for the various methods of segmentation. The pulse coupled neural network and k-means clustering segmentation methods failed considerably and therefore are not considered in the above table. In addition to better segmentation accuracy the seed pruning algorithm also has better false negative and false positive ratios compared to the other methods.

### 9.5 Region of Convergence curves based on varying thresholds

We also conducted experiments to determine the appropriate thresholds used in the major steps – a) determining Candidate seed pixels at each resolution level in the multi-resolution pyramid, b) the percentage of candidate seed neighbors required for a potential

seed pixel to survive the pruning step, c) the homogeneity criterion in the region growing scheme [8] and d) the closing window size for the post-processing step.

### 9.5.1 Candidate Seed Pixel Selection Threshold

The second step in the seed pruning algorithm (Chapter. 6.) describes the selection of *Candidate seed pixels* using a threshold. This threshold is varied and the number of *final seed pixels* selected as a result of the seed pruning algorithm is plotted against the threshold values. The graph shown in Fig.39. uses the logarithmic scale. Also the effect of various thresholds on segmentation accuracy [3] and false negative and false positive ratios [23] is plotted and is seen in Fig. 40.

The threshold which gives us the best possible accuracy while also limiting the number of seed pixels inputs to a feasible number is 160. This is chosen as the threshold for this step. As evident from Fig. 39, the number of seeds is zero for lower values of threshold, for all the images, or most. In cases, where the seeds are zero, no segmentation occurs, and consequently accuracy is 0%, FNR is 100%, and FPR is 0%. In cases, where few images showed seeds, a reliable measure across all images couldn't be established and hence the values for the few images are shown, but since it is not a reliable measure, it is shown by the dotted lines in Fig. 40.

The solid lines of the graph in Fig. 40, represent accurate values of accuracy, FNR and FPR averaged over all images. The dotted lines for higher threshold values, are estimated values due to unfeasibly high number of seed pixels surviving, making it virtually impossible to perform region growing.

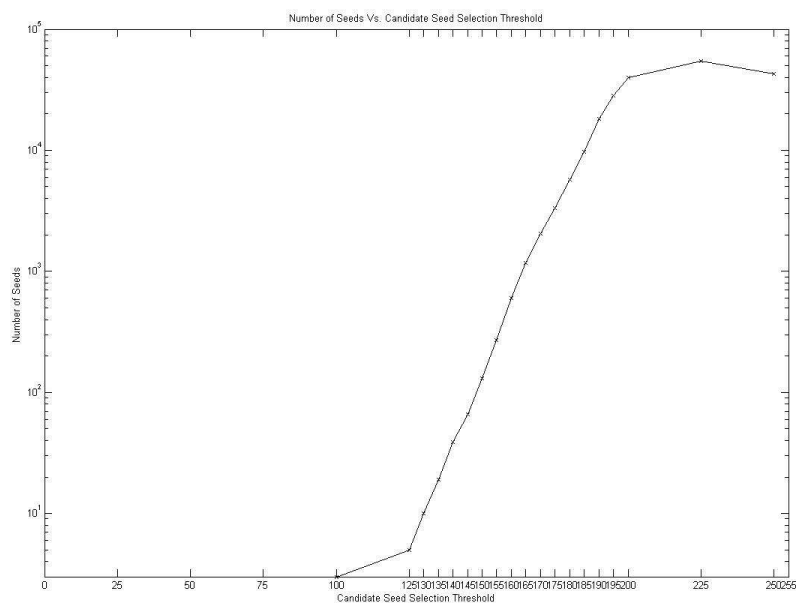


Figure 39: Number of *Final seed pixels* Vs. *Candidate seed pixel* selection threshold.

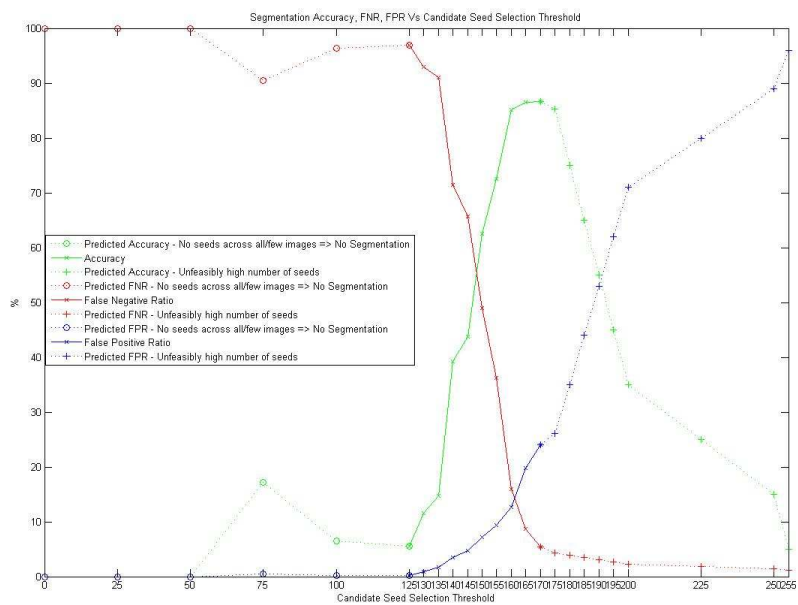


Figure 40: Accuracy, FNR and FPR Vs. *Candidate seed pixel* selection threshold.

### 9.5.2 Percentage of Candidate seed pixel neighbors

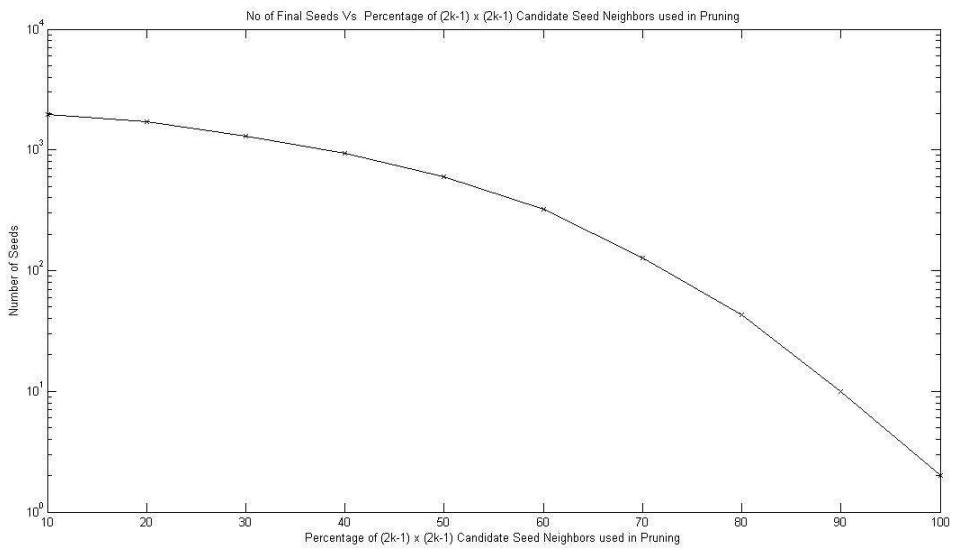


Figure 41: Number of *Final seed pixels* Vs. *Candidate seed pixel* neighbor percentage.

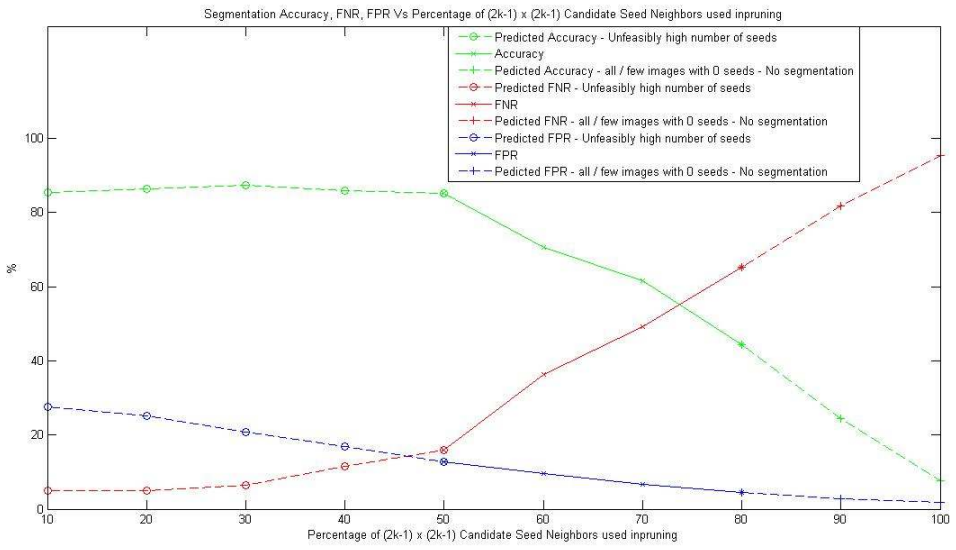


Figure 42: Accuracy, FNR and FPR Vs. *Candidate seed pixel* neighbor percentage.

In step 6. of the seed pruning algorithm (Chapter 6.), the percentage of *candidate seed pixel* neighbors of a *potential seed pixel* determines whether it survives the pruning

step and becomes a *fixed seed pixel*. Accordingly these values were varied and plots of number of *final seed pixels* selected by the seed pruning algorithm and segmentation accuracy, FNR and FPR values are shown in Figs. 41 and 42 respectively. The value of 50% is chosen. Intuitively, a value of 50% implies a higher probability of *candidate seed pixel* neighbors being present in at least two directions surrounding the *potential seed pixel* increasing the possibility of selection of seed pixels closer to the center of the ROI, eliminating border pixels which may give rise to inaccurate segmentations.

The value of 50% is also the lowest at which all images showed a set of *final seed pixels* selected, resulting in segmentations. Values below 50% are values from fewer images.

Segmentation accuracy drops as the percentage increase, as it damps the seed pruning algorithm to prune larger numbers of seeds resulting in pruning of accurate seed pixels too. Consequently FNR and FPR are also affected.

### **9.5.3 Homogeneity Criterion**

The homogeneity criterion threshold of the region growing segmentation scheme is also varied and the accuracy, FNR and FPR values are studied in Fig. 43. The homogeneity criterion imposes a bi-level threshold obtained at ten percent on either side of the homogeneity criterion threshold value varied here. Pixels satisfying the homogeneity criterion are expected to be part of an accurate ROI. Consequently, this is an important parameter on which accuracy is dependant.



Since this is not a seed selection parameter, the number of seeds selected remain constant across the threshold range, and is dependant only on the parameters discussed in Section 9.5.2 and Section 9.5.3.

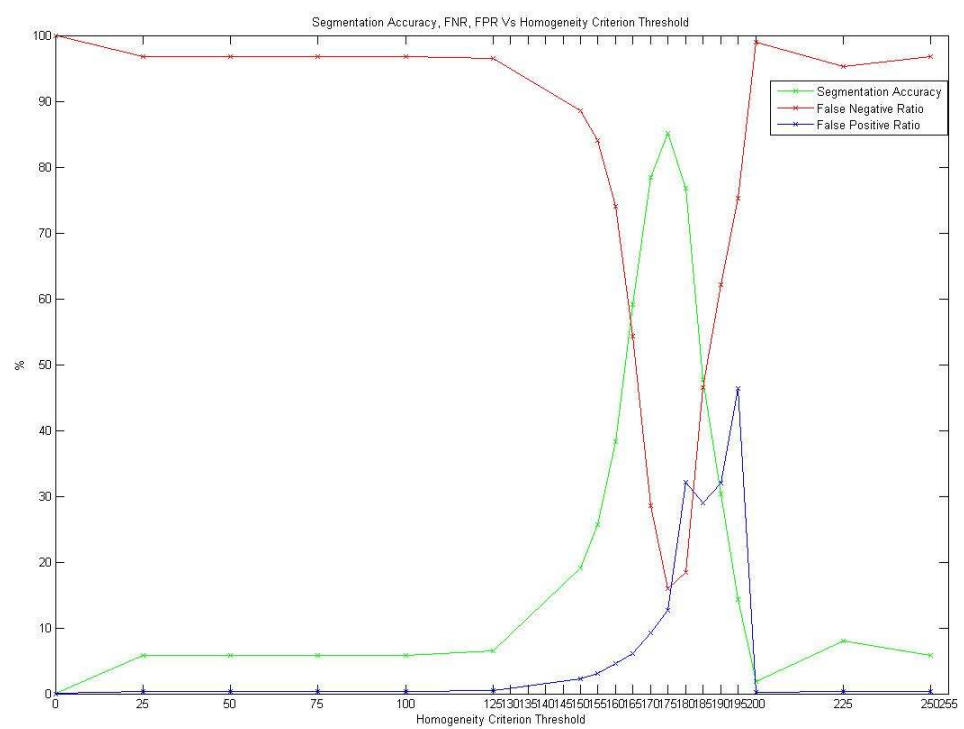


Figure 43: Accuracy, FNR and FPR Vs. Homogeneity criterion.

### 9.5.3 Morphological closing window size

Due to the inherent complexities in the manual segmentation (namely small non-ROI hole pixels appearing as ROIs due to inability to accurately manually segment the images, a post processing step that involves a morphological closing operation [3] to

smooth the holes present in the ROIs, is used before accuracy and false negative and false positive ratio calculations. The size of the window is varied, and the accuracy, FNR and FPR values are plotted against it in Fig. 44.

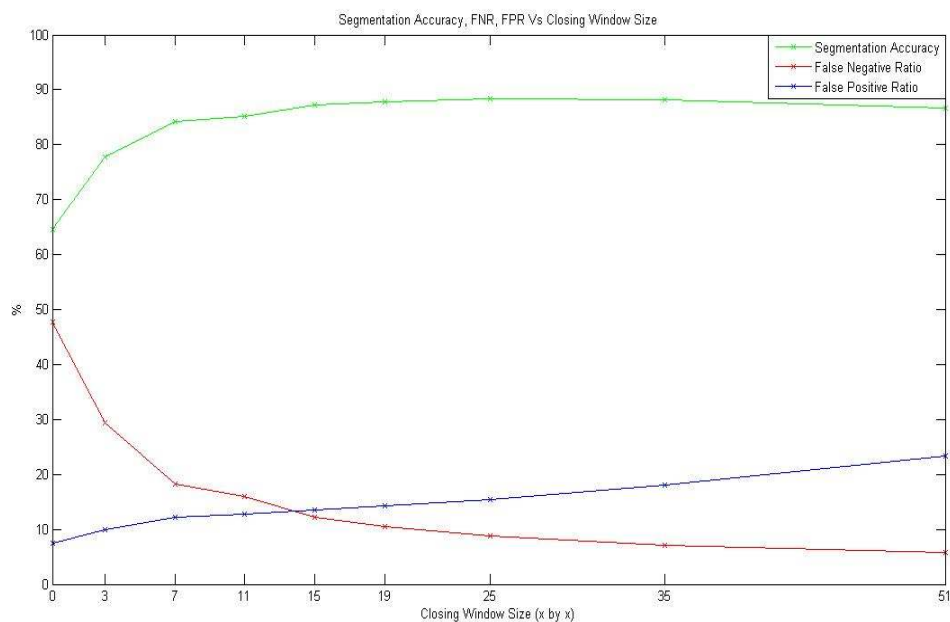


Figure 44: Accuracy, FNR and FPR Vs. Closing Window size.

The closing window size selected is  $11 \times 11$ . Even though this may seem too large, on high-resolution images of  $1600 \times 3200$  pixels, the window size is appropriate. Increasing window size, increases accuracy by reducing the FNR introduced by the non-ROI hole pixels within the manual segmentation, but does so at the expense of FPR values.

## Chapter 10

### 10. IMAGE REGISTRATION

Chapters 10. and 11. describe additional work done in order to complete the project, namely registration of the images and the creation of a 3-D model for visualization purposes.

Image registration is the process of overlaying two or more images of the same scene taken at different times, from different viewpoints, and/or by two sensors. It geometrically aligns two images – the *reference* and *sensed* images. The present differences between images are introduced due to different imaging conditions, in our project, through different slices in breast cancer tissue. The intra-slice resolution is extremely small (of the order of microns) thereby ensuring that there is not vast differences in the data across slices.

Zitova and Flusser [24] have given us a survey of different image registration methods. The majority of the registration methods consist of the following four steps:

a) *Feature Detection*: Salient and distinctive objects (closed-boundary regions, edges, contours, line intersections, corners etc.) are manually, or preferably, automatically detected. For further processing, these features can be represented by their point representatives (centers of gravity, line endings, distinctive points) which are called control points in the literature.

b) *Feature Matching*: In this step, the correspondence between the features detected in the sensed image and those detected in the reference image is established. Various feature descriptors and similarity measures along with spatial relationships among the features are used for that purpose.

c) *Transform Model Estimation*: The type and parameters of the so-called mapping functions, aligning the sensed image with the reference image, are estimated. The parameters of the mapping functions are computed by means of the established feature correspondence.

d) *Image Resampling and Transformation*: The sensed image is transformed by means of the mapping functions. Image values in non-integer coordinates are computed by the appropriate interpolation technique.

There are three major types of difference transformations between reference and sensed images. They are, *scaling differences*, *translational differences*, and *rotational differences* [24]. Since the images involved in our project are slices through breast cancer tissue, the major transformations are translational and rotational. Any scaling differences that occur between slices are actual changes in the tissue structure and we must not attempt to correct them while performing the registration.

We used an area-based method for feature detection and feature matching [24]. These methods deal with the images without attempting to detect salient objects. While a disadvantage of using an area-based method for registration purposes is that there is a

high probability that a window containing a smooth area without any prominent details will be matched incorrectly with other smooth areas in the reference image due to its non-saliency, it is perfect for our problem of simple rotational and translational registration.

## 10.1 Cross Correlation

One of the simplest pixel/voxel similarity measures is the sum of squared intensity differences between images (SSD), which is minimized during registration [24]. SSD is an optimum measure when two images only differ by Gaussian noise which is mostly the case for monomodality registrations. This assumption will not be invalid for multimodality images because of the intrinsic differences in the pixel/voxel intensities obtained using the two modalities.

$$SSD = \frac{1}{N} \sum_x (T(x) - S(t(x)))^2 \quad (10.1.1)$$

Where N gives the number of pixels in the image, T(x) represents the sensed image, S(x) represents the reference image which has to be registered with the sensed images and t(x) gives the transformation applied on the reference image.

The SSD measure is very sensitive to a small number of pixels/voxels that have very large intensity differences between images A and B. The effects due to large intensity

differences can be reduced by using the sum of absolute differences, SAD rather than SSD. SAD is given by:

$$SAD = \frac{1}{N} \sum_x |T(x) - S(t(x))| \quad (10.1.2)$$

The SSD measure makes the implicit assumption that after registration, the images differ only by Gaussian noise [24]. A slightly less strict assumption would be that, at registration, there is a linear relationship between the intensity values in the images. In this case, the optimum similarity measure is the correlation coefficient, CC.

$$CC = \frac{\sum_x (T(x) - \bar{T}) \cdot (S(t(x)) - \bar{S})}{\sqrt{\sum_x (T(x) - \bar{T})^2 \cdot \sum_x (S(t(x)) - \bar{S})^2}} \quad (10.1.3)$$

Here  $\bar{T}$  and  $\bar{S}$  gives the average intensity values in the sensed and reference image respectively. This measure of similarity is generally computed for window pairs from the reference and the sensed images and its maximum is searched. The window pairs for which the maximum is achieved are set as the corresponding ones. Although the CC based registration can exactly align mutually translated images only, it can also be successfully applied when slight rotation and scaling are present. Classical intensity based methods like sum of squared differences (SSD) and cross-correlation (CC) exploit

for matching directly image intensities, without any structural analysis. Consequently, they are sensitive to the intensity changes, introduced for instance by noise, varying illumination, and/or by using different sensor types.

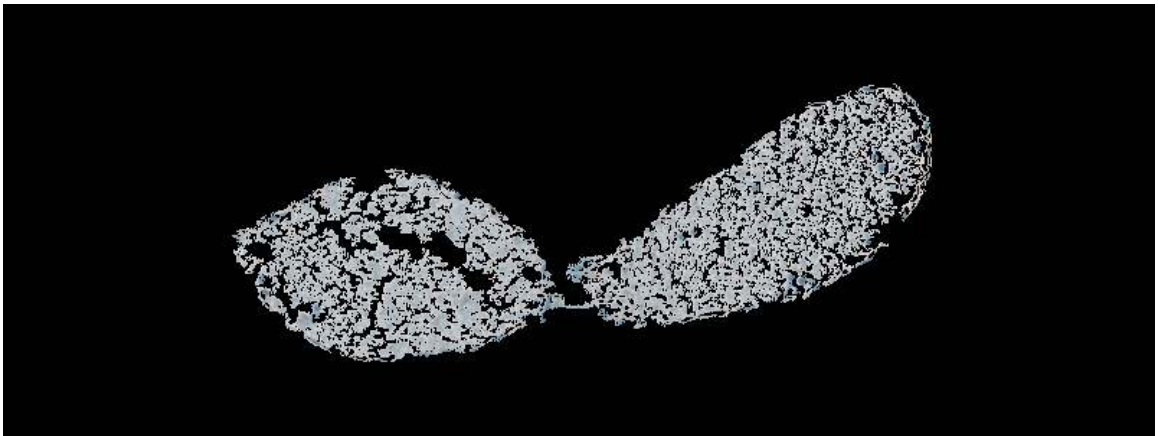


Figure 45: Reference Image.

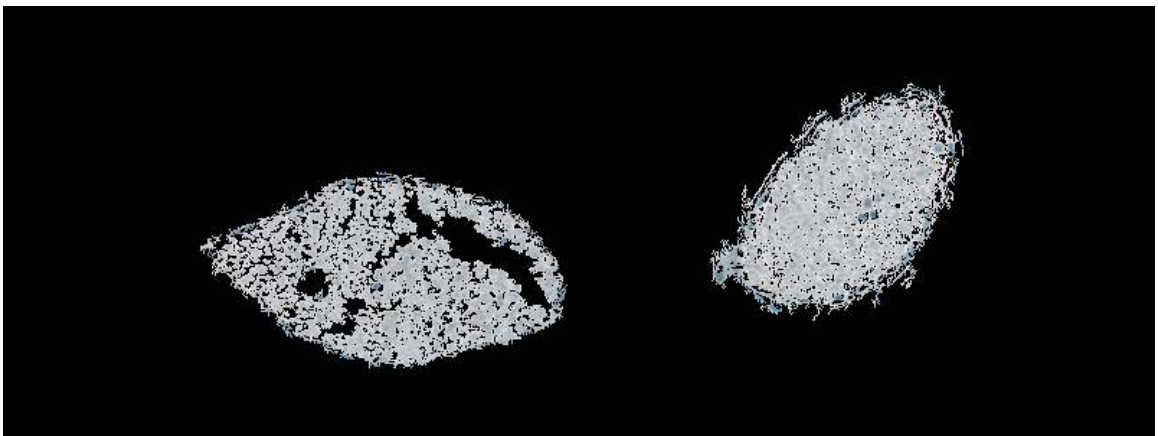


Figure 46: Sensed Image before registration.

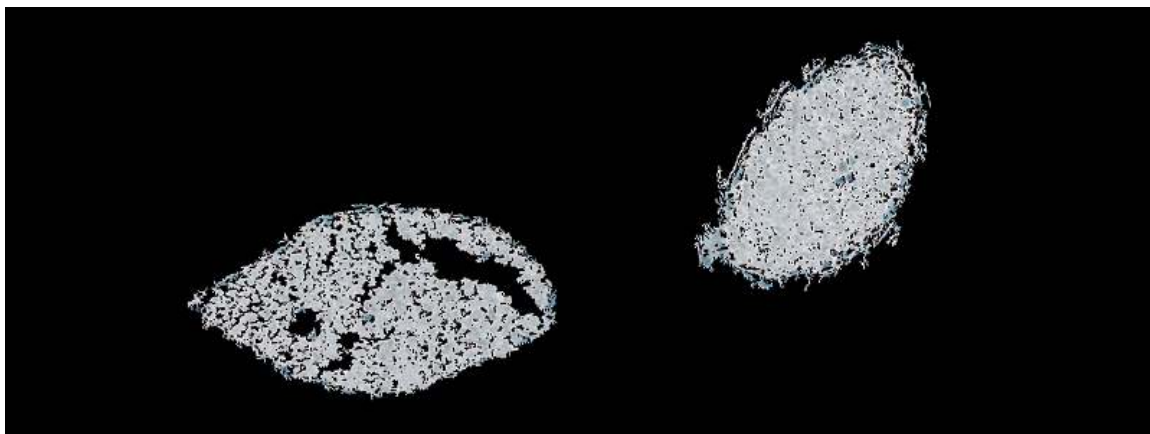


Figure 47: Sensed Image registered to Reference Image.



## **Chapter 11**

### **11. 3-D MODEL CREATION**

The final part of the thesis deals with the ultimate goal of the research project – the creation of the 3-D model of the vasculature from individual tissue slices that have been segmented using the Seed Pruning algorithm described in chapter 6. This is accomplished using commercially available software called Slicer Dicer. The registered images are arranged in order of slices and converted into a data matrix format such as .hdf and .netcdf which is read by the Slicer Dicer software and the 3-D model is presented for visualization purposes. The software then allows us to visualize the segmented vasculature in individual breast cancer tissue slices as a single 3-D object (as shown in Fig. 48).

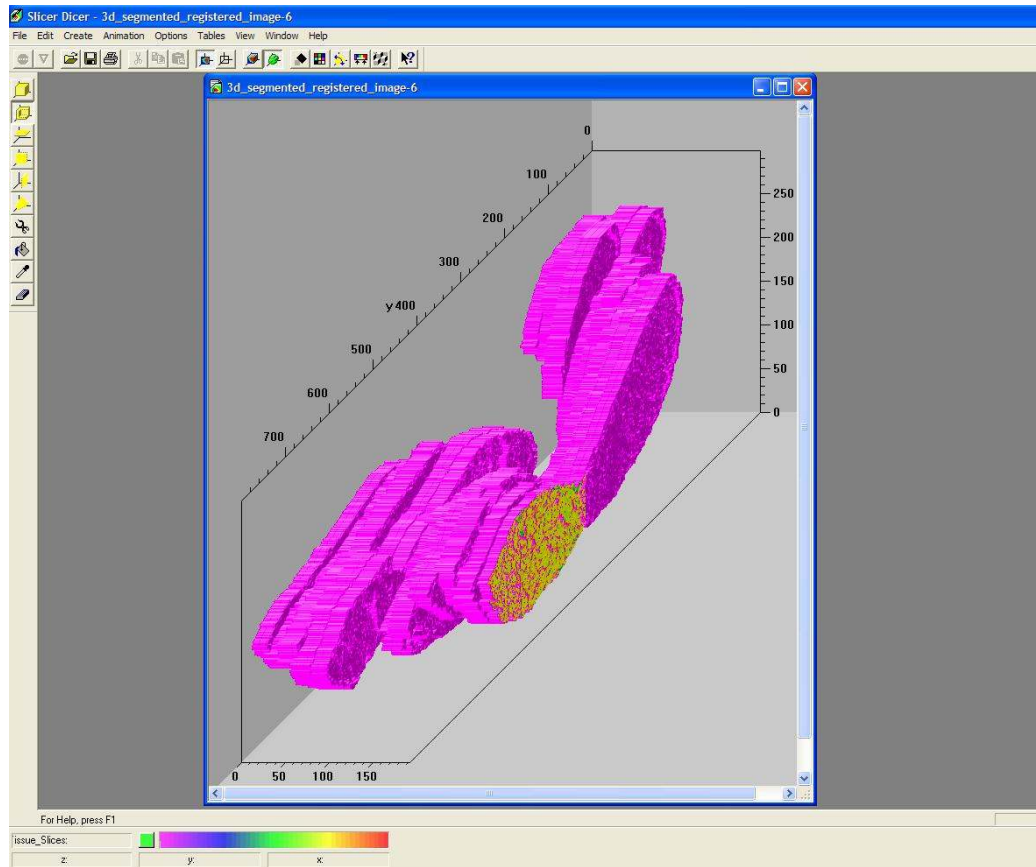


Figure 48: 3-D model of segmented data viewed using Slicer Dicer.

It also enables us to view the 3-D model from various angles and rotations and as individual slices in all three dimensions as well as oblique slices. Transparency, rotations and scaling can also be performed as shown in Fig. 49.

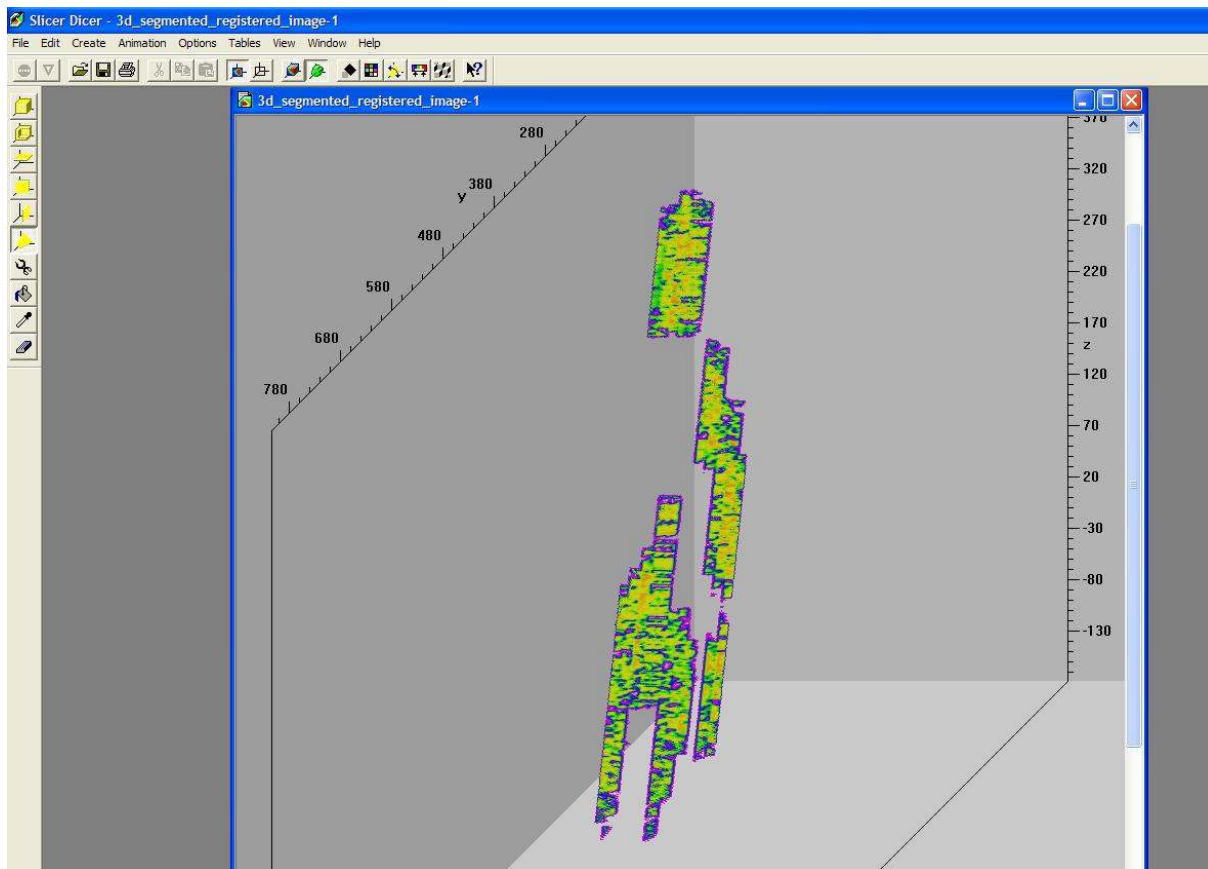


Figure 49: Oblique slice passing through an oblique plane, showing data components.

## Chapter 12

### 12. CONTRIBUTIONS AND FUTURE WORK

#### 12.1 Contributions

The principal goal of this thesis was to automate the manual segmentation of ROIs (vasculature) from high-resolution digitized images of immunohistochemically stained breast cancer tissue images.

Various segmentation models were implemented but their irreproducibility across datasets comprising of poorer stains due to different staining ‘runs’ (stained at different times) resulting in differences in stain retention by the ROIs necessitated research to develop a segmentation scheme to automatically segment these datasets. Many state of the art segmentation schemes like color profiling [1], 3-D histograms [15] and pulse coupled neural networks [16] were implemented, and the region growing scheme [8] was found to work best in segmenting the ROIs with an acceptable level of accuracy. However, due to the very high resolution of these images, the number of seed pixels needed for the region growing algorithm was extremely high, and using the region growing scheme as such was extremely time consuming and computationally complex. This necessitated the implementation of a scheme that would select fewer number of seed pixels to be used as the initial inputs for the region growing scheme. Even though many state of the art seed selection schemes like the integrated color edge method [14] and the histogram peak selection method [19] were found to minimize the number of seed pixels,

they either didn't achieve a feasible minimization in the number or did so at the expense of segmentation accuracy or both.

The major contribution of this research is the seed pruning algorithm which is a method for optimal selection of a minimal number of seed pixels which, when used as inputs to a region growing segmentation algorithm results in accurate automated segmentations of digitized high-resolution images of immunohistochemically stained breast cancer tissue vasculature, showing poorer stain retention.

The proposed seed pruning algorithm used principal component analysis [20] to generate a grayscale image retaining the important feature distinguishing chrominance information, and created a Gaussian multi-resolution pyramid [22], comprising  $n$  levels. A simple neighbor pixel search method was employed to finally prune the number of seed pixels in the original resolution image (20X) down to a feasible number starting from the base of the pyramid (2.5X) and moving upwards through the  $n=4$  resolution levels, reducing the number of surviving seed pixels at each resolution level.

The final number of seed pixels obtained at the original resolution level (20X) is much smaller than other seed pixel selection methods and still yields acceptable segmentations. The seed pruning algorithm facilitates region growing in high-resolution images where there is an inherent need to reduce the number of seed pixels used.

The automated segmentation was compared to the manual segmentation, and a mutual overlap performance metric was used to quantify the segmentation accuracy.

The seed pruning algorithm was compared with a pulse-coupled neural network scheme in terms of accuracy, and with a color edge based scheme and a histogram-based

peak selection algorithm in terms of accuracy and percentage reduction in number of seed pixels selected, and was found to be significantly better.

The effect of changing resolution levels at the base of the multi-resolution pyramid on number of seed pixels selected by the seed pruning algorithm and accuracy of corresponding segmentations has been discussed.

The false positive and false negative ratios [23] of the seed pruning segmentation was computed and was also found to be significantly better compared with the other methods.

## **12.2 Future Work**

While the seed pruning algorithm achieves appreciable minimizations in the number of seed pixels over other methods, there is always scope for improvement. Depending on the nature of the high resolution images involved in the project, the number of seed pixels selected may still be too high, and it can be further reduced by implementing a scheme where pixels at a higher resolution level is added to the set of potential seed pixels at that resolution level, and could be used in actively reducing the set of fixed seed pixels at that level.

The accuracy cross-correlation method involved in the registration step, currently depends on the image parameters, and is time consuming depending on the confidence interval window chosen. Algorithms capable of registering the images using non-window parameters could be applied to this area.

## Chapter 13

### 13. REFERENCES

- [1] N. Duffy, J. Crowley, and G. Lacey, "Object detection using colour," *IEEE Conf. on Pattern Recognition*, vol. 1, pp. 700-703, Sep. 2000.
- [2] S.J. Sangwine and R.E.N. Horne, *The Colour Image Processing Handbook*, 1st ed. Chapman & Hall, 1998.
- [3] M. Sonka, V. Hlavac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd ed. Thomson, Brooks/Cole, 2001.
- [4] P.K. Sahoo, S. Soltani, and A.K.C. Wong, "A survey of thresholding techniques," *Comput. Vis. Graph. Image Process.*, vol. 41, pp. 233-260, 1988.
- [5] N. Pal and S. Pal, "A review on image segmentation techniques," *Pattern Recognit.*, vol.26, pp. 1277-1294, 1993.
- [6] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: active contour models," in *Proc. 1<sup>st</sup> ICCV*, 1987, pp. 259-267.
- [7] C. Xu and J.L. Prince, "Snakes, shapes and gradient vector flow," *IEEE Trans. on Image Processing*, vol. 7, no. 3, pp. 359-369, Mar. 1998.
- [8] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 641-647, Jun. 1994.
- [9] R. M. Haralick and L.G. Shapiro, "Survey: Image segmentation techniques," *Comput. Vis. Graph. Image Process.*, vol. 29, pp. 100-132, 1985.

- [10] S.A. Hojjatoleslami and J. Kittler, "Region growing: A new approach," *IEEE Trans. Image Processing*, vol. 7, pp. 1079-1084, 1998.
- [11] T. Pavlidis and Y. T. Liow, "Integrating region growing and edge detection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 225-233, 1990.
- [12] J. Haddon and J. Boyce, "Image segmentation by unifying region and boundary information," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 12, pp. 929-948, 1990.
- [13] S.C. Zhu, T.S. Lee and A.L. Yuille, "Region competition: Unifying snakes, region growing, Energy/Bayes/MDL for multi-band segmentation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 9, pp. 884-900, Sep. 1996.
- [14] 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 Trans. on Image Processing*, vol. 10, no. 10, pp. 1454-1466, Oct. 2001.
- [15] A. Gaddipati, J.F. Cornhill, E. Herderick and R. Yagel, "An efficient method for automated segmentation of histochemically stained slides," *IEEE Conf. on Engineering in Medicine and Biology Society*, vol. 1, pp. 487-498, Sep. 1995.
- [16] R.D. Stewart, I. Fermin, and M. Opper, "Region growing with pulse-coupled neural networks: an alternative to seeded region growing," *IEEE Trans. on Neural Networks*, vol. 13, no. 6, pp. 1557-1562, Nov. 2002.
- [17] T. Kanungo, D.M. Mount, N.S. Netanyahu, C.D. Piatko, R. Silverman, and A.Y. Wu, "An efficient k-means clustering algorithm: analysis and implementation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 881-892, Jul. 2002.
- [18] A.K. Jain, and R.C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.



- [19] A.O. Silva, J.F.C. Wanderley, A.N. Freitas, H.D.F Bassani, R.A. de Vasconcelos and F.M.O. Freitas, "Watershed transform for automatic segmentation of the human pelvis area," *IEEE Intl. Conf. on Acoustics, Speech and Signal Processing*, 2004, vol. 5 – pp. 597-600.
- [20] R.O. Duda, P.E. Hart, and D.G.Stork, *Pattern Classification*, 2nd ed. Wiley Interscience, 2001.
- [21] J.-C. Devaux, P. Gouton and F.Truchetet, "Aerial colour image segmentation by Karhunen-Loeve Transform," *IEEE Intl. Conf. on Pattern Recognition*, 2000, vol. 1 –pp. 309-312.
- [22] P.J. Burt and E.H. Adelson, "The Laplacian Pyramid as a Compact Image Code," *IEEE Trans. on Communications*, vol. COM-31, no. 4, pp. 532-540, Apr. 1983.
- [23] T. Liang and J.J. Rodríguez, "MR cranial image segmentation-a morphological and clustering approach," *Proc. of the IEEE Southwest Symp. On Image Analysis and Interpretation*, 1996, pp. 184-189.
- [24] B.Zitova and J. Flusser, "Image registration methods: a survey", *Elsevier Image and Vision Computing*, vol. 21, pp. 977-1000, 2003.