

**DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES**

**ANALYSIS AND PROCESSING OF
HISTOPATHOLOGICAL IMAGES**

by
Sibel BARDAKÇI

**August, 2012
İZMİR**

ANALYSIS AND PROCESSING OF HISTOPATHOLOGICAL IMAGES

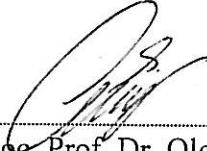
**A Thesis Submitted to the
Graduate School of Engineering and Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of Science
in Electrical and Electronics Engineering**

**by
Sibel BARDAKÇI**

**August, 2012
İZMİR**


M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled "ANALYSIS AND PROCESSING OF HISTOPATHOLOGICAL IMAGES" completed by SİBEL BARDAKÇI under supervision of ASSOC. PROF. DR. OLCAY AKAY and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.



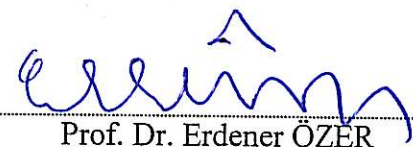
Assoc. Prof. Dr. Olcay AKAY

Supervisor



Asst. Prof. Dr. Güleser Kalaycı DEMİR

(Jury Member)



Prof. Dr. Erdener ÖZER

(Jury Member)



Prof. Dr. Mustafa SABUNCU
Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

First and foremost, I would like to express my respect and gratitude to my supervisor Assoc. Prof. Dr. Olcay AKAY. I must express that his scholarship and attention to details have set an example for me and added considerably to my graduate experience.

I also would like to thank Asst. Prof. Dr. Gülezer Kalaycı DEMİR for her valuable help in determination of the thesis topic, for her crucial advice, and important contributions.

I gratefully acknowledge Prof. Dr. Erdener ÖZER for his valuable contributions and recommendations during my thesis study.

I also would like to thank Asst. Prof. Dr. M. Alper SELVER for his support, crucial advice, and valuable contributions.

Last but not the least, I would like to thank my family for supporting me spiritually throughout my life.

Sibel BARDAKÇI

ANALYSIS AND PROCESSING OF HISTOPATHOLOGICAL IMAGES

ABSTRACT

Neuroblastoma (NB) is a cancer of nerve cell origin commonly affecting infants and children. For treatment planning of the tumor, histopathological examinations performed by expert pathologists are required to characterize the histology. This analysis guides the experts to determine the histological character stage and gives information about treatment methods.

Unfortunately, the qualitative visual examination performed by pathologists under the microscope is tedious and prone to error due to several factors. First, for NB diagnosis, pathologists typically pick some representative regions at lower magnifications (e.g. 2×, 4×) and examine only those regions. The final decision about the entire slide is based on these sampled regions. Although this approach provides accurate decisions, it may be misleading for heterogeneous tumors. Second, the resulting diagnosis varies considerably between different examiners. Experience and fatigue may cause variations among pathologists.

The purpose of this thesis is to develop an algorithm by using image processing and classification techniques and decrease the decision variations to the lowest level by simplifying the diagnosis for the pathologists as much as possible. For this goal, images belonging to the tissue samples of NB taken from the patient are examined histologically and analyzed. The images with different magnifications are captured by using the electron microscope. The percentage of neuropil is calculated and the tumor cells are determined by using these captured images. After the application of various image processing techniques, the feature matrices are created by using the extracted region and texture features. Then, via these feature matrices, classification of the cells are performed with the help of artificial neural networks (ANNs) and some other machine learning algorithms (ensemble methods). Depending on these classification results, mitosis karyorrhexis (MK) index and the grade of differentiation are determined. Thanks to these calculations, a computer-based tool is

provided to pathologists for determining the stage of NB disease from the tissue images.

Keywords: Neuroblastoma, image processing, neural networks, Haralick texture features, ensemble methods, morphological operations, resampling methods, dimensionality reduction, cross validation, bootstrap.

HİSTOPATOLOJİK GÖRÜNTÜLERİN ANALİZİ VE İŞLENMESİ

ÖZ

Nöroblastom çoğunlukla bebekleri ve çocukları etkileyen nöral hücre kökenli bir kanserdir. Tümörün tedavi planlaması için patoloğlar tarafından histopatolojik inceleme gerekmektedir. Bu histolojik analiz hastalık evresini tespit etmek ve tedavi yöntemleri hakkında bilgi vermek için uzman hekimlere rehberlik eder.

Ne yazık ki patoloğlar tarafından mikroskop altında yapılan nitel görsel inceleme uzun süren bir işlemdir ve çeşitli faktörlerden dolayı hata yapmaya yatkındır. Öncelikle, nöroblastom teşhisi için, patoloğlar genel olarak daha düşük büyütme oranlarında (2x, 4x, vs.) bazı belirgin bölgeleri alırlar. Tüm slayt hakkındaki son karar bu örnek bölgelere bağlıdır. Bu yaklaşım doğru karar verilmesini sağlasa da heterojen tümörler için yanıltıcı olabilir. İkinci olarak, son teşhis farklı kişilerin incelemesine göre önemli ölçüde farklılık gösterebilmektedir. Deneyim ve yorgunluk patoloğlar arasında önemli farklılıklara neden olabilir.

Bu tezin amacı, görüntü işleme ve sınıflandırma teknikleri kullanarak bir algoritma geliştirmek ve bu algoritma ile teşhisi patoloğlar için mümkün olduğunca kolaylaştırarak kişisel hataları en aza indirmektir. Bu amaçla, nöroblastom hastasından alınan doku örneğine ait görüntüler histolojik olarak incelenmekte ve analiz edilmektedir. Nöroblastom doku örneğinden elektron mikroskobu yardımı ile farklı büyütme oranlarında görüntüler alınmaktadır. Bu alınan görüntüler kullanılarak nöropil yüzdesi hesaplanır ve tümör hücreleri tespit edilir. Çeşitli görüntü işleme teknikleri uygulandıktan sonra resimlerden çıkarılan bölgesel ve dokusal öznitelikler ile öznitelik matrisleri oluşturulur. Daha sonra bu öznitelik matrisleri yapay sinir ağları ve diğer bazı otomatik öğrenme algoritmalarına giriş olarak kullanılarak hücrelerin sınıflandırılması sağlanmaktadır. Bu sınıflandırma sonuçlarına bağlı olarak mitosis karyoreksis (MK) indeksi ve diferansiyasyon derecesi hesaplanmaktadır. Bu sayede nöroblastom doku görüntülerinden hastalık safhası tespit edilerek patoloğlara yardımcı bir araç sağlanmaktadır.

Anahtar kelimeler: Nöroblastom, görüntü işleme, yapay sinir ağıları, Haralick doku öznelikleri, ensemble yöntemleri, morfolojik işlemler, yeniden örnekleme metodları, boyut azaltma, çapraz doğrulama, bootstrap.

CONTENTS

	Page
M.Sc THESIS EXAMINATION RESULT FORM.....	ii
ACKNOWLEDGEMENTS	iii
ABSTRACT.....	iv
ÖZ	vi
CHAPTER ONE - INTRODUCTION	1
1.1 The Studies for Neuroblastoma Classification	3
1.2 Outline of the Thesis	5
CHAPTER TWO - THEORETICAL BACKGROUND ON NEUROBLASTOMA	6
2.1 Neuroblastic Tumors	6
2.1.1 Neuroblastoma	6
2.1.1.1 Grade of Differentiation.....	7
2.1.1.2 Mitosis Karyorrhexis (MK) Index	9
CHAPTER THREE - THEORETICAL BACKGROUND ON IMAGE PROCESSING TECHNIQUES	10
3.1 Perception-Based Color Spaces.....	10
3.1.1 HSV Color Space	10
3.2 Segmentation	11

3.3 Morphological Operators	12
3.3.1 Dilation and Erosion	13
3.3.2 Opening and Closing	14
3.4 Feature Extraction	14
3.4.1 Low Level Feature Extraction	15
3.4.2 Object Description	15
3.4.3 Texture Description	16
3.4.3.1 Statistical Approaches	17
3.4.4 Dimensionality Reduction	20
3.4.4.1 Principal Component Analysis (PCA)	20
3.4.4.2 Feature Selection	20

CHAPTER FOUR – THEORETICAL BACKGROUND ON ARTIFICIAL NEURAL NETWORKS 22

4.1 Computational Models of Neurons	22
4.2 Network Architectures.....	23
4.3 Learning.....	24
4.4 Resampling Techniques.....	26
4.4.1 Cross Validation.....	26
4.4.1.1 M-Fold Cross Validation	26
4.4.1.2 Leave-One-Out Cross Validation.....	27
4.4.2 Bootstrap	27
4.4.3 Boosting	28
4.4.3.1 AdaBoost.....	28

CHAPTER FIVE - THE EXPERIMENTAL STUDY.....	30
5.1 Detection of Neuropil Regions.....	32
5.2 Detection of Cells in Differentiated Images.....	38
5.3 Detection of Cells in the Images for Determining Mitosis Karyorrhexis (MK)	
Index.....	48
5.4 Feature Extraction.....	57
5.5 Classification.....	62
5.5.1 Input Data for Classification.....	62
5.5.2 Artificial Neural Network.....	67
5.5.2.1 Cross Validation.....	68
5.5.2.2 Bootstrap.....	69
5.5.2.3 Random Selection.....	69
5.5.2.4 Increasing the Number of Weak Class.....	69
5.5.3 Ensemble Methods.....	70
5.5.3.1 Adaptive Boosting.....	70
5.5.3.2 Robust Boosting.....	70
5.5.4 Dimensionality Reduction.....	71
5.5.4.1 Normal Training.....	71
5.5.4.2 Principal Component Analysis (PCA).....	71
5.5.4.3 Feature Selection.....	71
5.6 Graphical User Interface (GUI).....	72
CHAPTER SIX - RESULTS.....	75
6.1 The Results of MK Index Subtype.....	75
6.2 The Results of Grade of Differentiation Subtype.....	80

6.3 The Results of Using Different Number of Neurons at ANNs	82
CHAPTER SEVEN - CONCLUSIONS	85
REFERENCES	87

CHAPTER ONE

INTRODUCTION

Through the increasing use of direct digital imaging systems for medical diagnostics, digital image processing becomes more and more important in health care. In addition to originally digital methods, such as Computed Tomography (CT) or Magnetic Resonance Imaging (MRI), initially analogue imaging modalities such as endoscopy or radiography are nowadays equipped with digital sensors. Digital images are composed of individual pixels, to which discrete brightness or color values are assigned. Based on digital imaging techniques, the entire spectrum of digital image processing is now applicable in medicine.

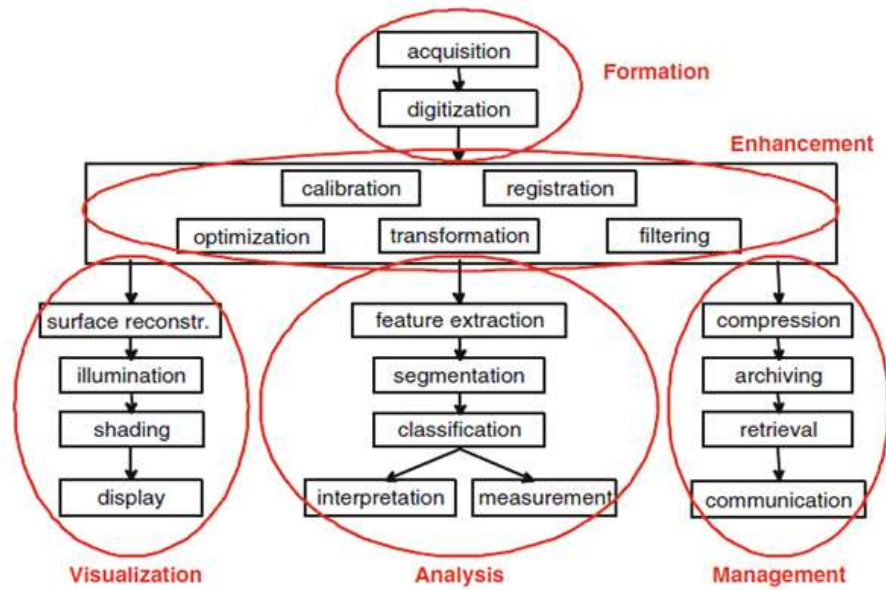


Figure 1.1 Modules of image processing (Deserno, T. M., 2011).

The commonly used term *biomedical image processing* means the provision of digital image processing for biomedical sciences. In general, digital image processing covers four major areas: image formation, image visualization, image analysis, and image management as seen in Figure 1.1 (Deserno, T. M., 2011).

Histopathological image analysis is an emerging field. It is becoming increasingly popular mostly due to the recent developments in the scanning technology, which

made it possible to digitize the whole tissue slides at high magnifications. However, there are several challenges ahead. The variations between samples of the same cancer type, either due to relatively distinct content or due to the slide preparation stages, make it difficult to develop adaptive and robust algorithms. Nevertheless, there are a number of computerized systems developed for several cancer types such as follicular lymphoma, prostate cancer, breast cancer, and neuroblastoma (Sertel, O., Catalyurek, U. V., Shimada, H., & Gurcan, M.N., 2009).

Neuroblastoma (NB) is a tumor that begins in nerve tissues in the neck, chest, abdomen, adrenal gland, or pelvis. About 50% of neuroblastomas start in the tissues of the adrenal glands, located just above the kidneys. Often this tumor spreads before it is diagnosed. The common sites are the lymph nodes, liver, bones, and bone marrow. Neuroblastoma occurs in early childhood with 2/3 of the children younger than 5 years of age when they are diagnosed.

The treatment depends on the extent and the nature of the tumor. Once a neuroblastoma is found, more tests are done to find out if it has spread to surrounding tissues or other parts of the body. This process is called staging.

- *Stage 1:* Tumor confined to the organ or structure of origin.
- *Stage 2:* Tumor extending beyond the organ/structure of origin, involving the lymph nodes on the same side of the tumor.
- *Stage 3:* Tumor extends beyond the midline, involves the lymph nodes on both sides of the body.
- *Stage 4:* Metastatic disease involving other parts of the body, especially the bones or bone marrow.
- *Stage 4S:* In a child younger than 12 months when there is evidence of liver, lymph node, or marrow involvement associated with a primary tumor which is often quite small.

Due to the large variation in neuroblastoma's morphological structure, the prognosis of this disease is challenging and affects the treatment plan. In current clinical practice, neuroblastoma classification is carried out by highly trained

pathologists with visual examinations of pathological slides under the microscope according to the International Neuroblastoma Classification System developed by Shimada H., Ambros, I. M., Dehner, L. P., Hata, J., Joshi, V. V., Roald, B., Stram, D. O., Gerbing, R. B., Lukens, J. N., Matthay, K. K. & Castleberry, R. P. (1999a). The Shimada classification system uses morphological information such as presence and absence of Schwannian cell development, the relative count of tumor cells in mitosis and karyorrhexis.

1.1 The Studies for Neuroblastoma Classification

Generally, pre-processing, segmentation, post-processing, feature extraction, and classification are applied to analyze the histopathology image. Pre-processing and post processing are used for enhancement of the image. Segmentation is used for detecting cell nuclei or cell components. Feature extraction is for obtaining the properties of the cells or cell components and classification is used for analyzing the data.

According to Shimada, et al. (1999a), neuroblastoma tissues are classified into two different subtypes. One of them is grade of differentiation (undifferentiated, poorly differentiated, well differentiated) and the other is mitosis karyorrhexis (MK) index (high, intermediate, low). There are different studies about classification methods of neuroblastoma tissues as seen below. The use of image processing methods simplifies and accelerates the analysis and prevents subjective results.

An automated cell nuclei segmentation method is developed by Gurcan, M. N., Pan, T., Shimada, H. & Saltz, J. (2006). This method employs morphological top-hat by reconstruction algorithm coupled with hysteresis thresholding to both detect and segment the cell nuclei. Accuracy of the automated cell nuclei segmentation algorithm is measured by comparing its outputs to manual segmentation. The average segmentation accuracy is $90.24 \pm 5.14\%$.

An automatic classification system is presented that includes a novel segmentation method using the Expectation Maximization (EM) algorithm with the Fisher-Rao

criterion as its kernel by Kong, J., Shimada, H., Boyer, K., Saltz, J. & Gurcan, M. N. (2007b). This is followed by a classification stage with classifiers applied to the actual neuroblastoma images. The good classification accuracy suggests that the developed method is promising in automating this pathological assessment.

Another method classifies the image either into low or high grades based on the amount of cytological components by Sertel, O., Kong, J., Lozanski, G., Shana'ah, A., Catalyurek, U., Saltz, J. & Gurcan, M. N. (2008a). To further discriminate the lower grades into low and mid grades, a novel color texture analysis approach was proposed. This approach modifies the gray level co-occurrence matrix method by using a non-linear color quantization with self-organizing feature maps (SOFMs). This is particularly useful for the analysis of Haematoxylin and Eosin (H&E) stained pathological images whose dynamic color range is considerably limited. Experimental results on real follicular lymphoma images demonstrate that the proposed approach outperforms the gray level based texture analysis.

An image analysis approach that operates on digitized NB histology samples is proposed by Sertel, O., Catalyurek, U. V., Shimada, H. & Gurcan, M. N. (2009). Based on the likelihood functions estimated from the samples of manually marked regions, a probability map is computed that indicates how likely a pixel belongs to mitosis and karyorrhexis cells. Component-wise 2-step thresholding of the generated probability map provides promising results in detecting mitosis and karyorrhexis cells with an average sensitivity of 81.1% and 12.2% false positive detections on average.

An image analysis system is proposed that operates on digitized H&E stained whole-slide NB tissue samples and classifies each slide as either stroma-rich or stroma-poor based on the degree of Schwannian stromal development by Sertel, O., Kong, J., Shimada, H., Catalyurek, U. V., Saltz, J. H. & Gurcan, M. N. (2008b). Their statistical framework performs the classification based on textural features extracted using co-occurrence statistics and local binary patterns. Due to the high resolution of digitized whole-slide images, a multi-resolution approach is proposed

that mimics the evaluation of a pathologist such that the image analysis starts from the lowest resolution and switches to higher resolutions when necessary. An offline feature selection step is employed, which determines the most discriminative features at each resolution level during the training step. A modified k-nearest neighbor classifier is used to determine the confidence level of the classification to make the decision at a particular resolution level. The proposed approach was independently tested on 43 whole-slide samples and provided an overall classification accuracy of 88.4%.

The development of a computer-aided system for the classification of grade of neuroblastic differentiation is developed by Kong, J., Sertel, O., Shimada, H., Boyer, K., Saltz, J. & Gurcan, M. N. (2007a). This automated process is carried out within a multi-resolution framework that follows a coarse-to-fine strategy. Additionally, a novel segmentation approach using the Fisher-Rao criterion, embedded in the generic EM algorithm, is employed. Multiple decisions from a classifier group are aggregated using a two-step classifier combiner that consists of a majority voting process and a weighted sum rule using priori classifier accuracies.

The aim of this thesis work is to find the number of mitosis and karyorrhexis cells to determine MK index and the percentage of neuropil and differentiated cells to determine the grade of differentiation in the neuroblastoma images. When the desired input data is obtained, the stage of the illness is automatically and objectively determined at a relatively short processing time.

1.2 Outline of the Thesis

The remainder of this thesis is organized as follows. Chapter Two outlines the fundamentals of neuroblastoma disease. Chapter Three and Chapter Four present the theoretical background of image processing techniques that are used in this study and theoretical background on artificial neural networks, respectively. The details of all the developed algorithms in this study are given step by step in Chapter Five. The obtained results of the developed algorithms are discussed in Chapter Six. In Chapter Seven, the results of the thesis are discussed and our conclusions are given.

CHAPTER TWO

THEORETICAL BACKGROUND ON NEUROBLASTOMA

2.1 Neuroblastic Tumors

Tumors of the neuroblastoma group (neuroblastic tumors), defined as embryonal tumors of the sympathetic nervous system, derive from the neural crest and arise in the adrenal medulla, paravertebral sympathetic ganglia, and sympathetic paraganglia, such as the organ of Zuckerkandl.

Neuroblastic tumors are assigned to one of four basic morphologic categories: *neuroblastoma* (Schwannian stroma-poor); *ganglioneuroblastoma, intermixed* (Schwannian stroma-rich); *ganglioneuroma* (Schwannian stroma-dominant); and *ganglioneuroblastoma, nodular* (composite Schwannian stroma-rich/stroma-dominant and stroma-poor).

2.1.1 Neuroblastoma

Neuroblastoma (NB) is a cancer of nerve cell origin. It commonly affects infants and children. Based on the American Cancer Society's statistics, it is by far the most common cancer in infants and the third most common type of cancer in children. As in most cancer types, histopathological examinations are required to characterize the histology of the tumor for further treatment planning. The World Health Organization recommends the use of the International Neuroblastoma Pathology Classification (the Shimada system) for categorization of the patients into different prognostic groups (Shimada, H., et al., 1999a, 1999b).

The grade of neuroblastic differentiation and mitosis karyorrhexis (MK) index are the most salient features that contribute to the final tissue classification as favorable and unfavorable histology.

The neuroblastoma classification is shown in Figure 2.1.

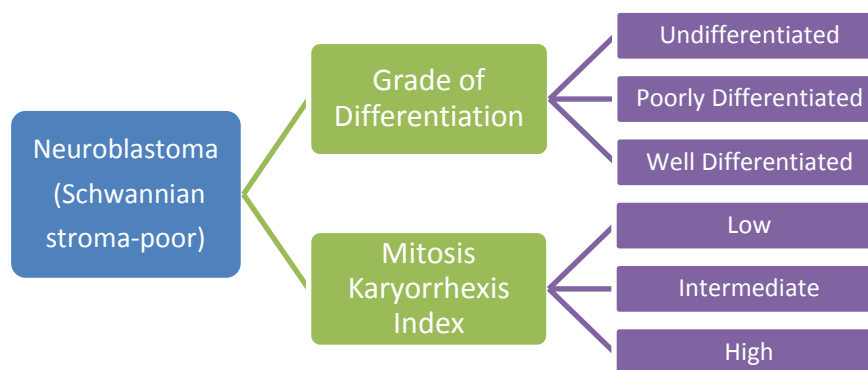


Figure 2.1 Neuroblastoma classification.

2.1.1.1 *Grade of Differentiation*

Neuroblastoma has three subtypes in terms of grade of differentiation; undifferentiated, poorly differentiated, and well differentiated as seen below.

Undifferentiated Subtype: Neuropil absent; no tumor cell differentiation; diagnosis relies heavily on ancillary techniques, such as immunohistochemistry, electron microscopy, and/or molecular/cytogenetics (see Figure 2.2).

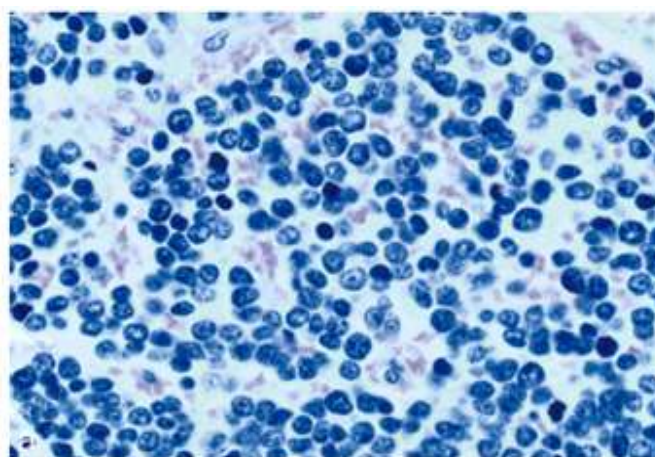


Figure 2.2 Neuroblastoma (Schwannian stroma-poor), undifferentiated subtype (Shimada, et al. 1999b).

Poorly Differentiated Subtype: Neuropil background evident; 5% or fewer tumor cells show a feature of differentiating neuroblasts with a synchronous differentiation of nucleus (enlarged, vesicular, with a single prominent nucleolus) and cytoplasm

(conspicuous, eosinophilic, or amphophilic, and 2 times larger in diameter than nucleus). The sample image of poorly differentiated subtype is shown in Figure 2.3.

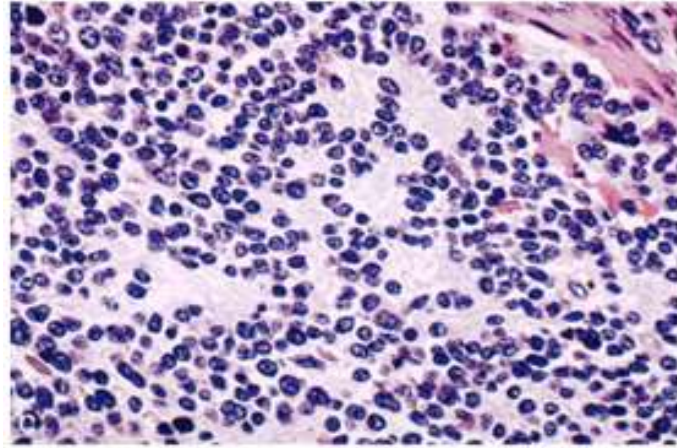


Figure 2.3 Neuroblastoma (Schwannian stroma-poor), poorly differentiated subtype, composed of undifferentiated neuroblastic cells with clearly recognizable neuropil (Shimada, et al. 1999b).

Well Differentiated Subtype: More than 5% of tumor cells show an appearance of differentiating neuroblasts (may be accompanied by mature ganglion-like cells), and neuropil is usually abundant. Some tumors can show substantial Schwannian stromal formation, frequently at their periphery, and a transition zone between neuroblastomatous and ganglioneuromatous region can develop, although this zone does not have well-defined borders and comprises less than 50% of the tumor (Qualman, S. J., Bowen, J., Fitzgibbons, P. L., Cohn, S. L. & Shimada, H., 2005). The well differentiated subtype image is shown in Figure 2.4.

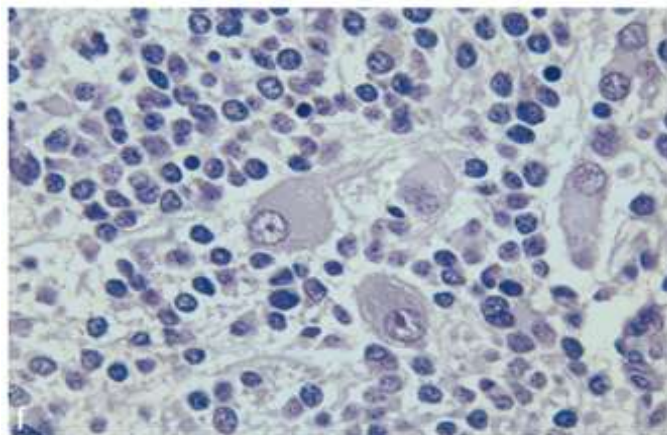


Figure 2.4 Differentiating neuroblasts in neuroblastoma (Schwannian stroma-poor), well differentiated subtype, showing synchronous nuclear and cytoplasmic differentiation (Shimada, et al. 1999b).

2.1.1.2 Mitosis Karyorrhexis (MK) Index

The MK index is the number of mitoses and karyorrhectic nuclei per 5000 neuroblastic cells. It is a useful prognostic indicator for tumors in the neuroblastoma (Schwannian stroma-poor) category and should be determined as an average of all tumor sections available.

- (1) *Low MK Index*: Fewer than 100 mitotic and karyorrhectic cells/5000 tumor cells, or less than 2% of tumor consisting of mitotic and karyorrhectic cells
- (2) *Intermediate MK Index*: 100 to 200 mitotic and karyorrhectic cells/5000 tumor cells, or 2% to 4% of tumor consisting of mitotic and karyorrhectic cells
- (3) *High MK Index*: More than 200 mitotic and karyorrhectic cells/5000 tumor cells, or greater than 4% of tumor consisting of mitotic and karyorrhectic cells (Shimada, et al. 1999b).

CHAPTER THREE

THEORETICAL BACKGROUND ON IMAGE PROCESSING TECHNIQUES

3.1 Perception-Based Color Spaces

Color spaces that are based intuitively on human color perception are of interest for the fields of computer vision and computer graphics. A color can be more easily described intuitively by values for hue, color saturation, and intensity than from vector components in the RGB (red, green, blue) or CMYK (cyan, magenta, yellow, black) color space.

3.1.1 HSV Color Space

In the HSV color space, colors are specified by the components hue, saturation, and value. Hue, saturation, and brightness values are used as coordinate axes. By projecting the RGB unit cube along the diagonals of white to black, a hexacone results that forms the topside of the HSV pyramid. The hue H is indicated as an angle around the vertical axis. Red is determined with $H = 0^\circ$ or $H = 360^\circ$, green with $H = 120^\circ$, and so on (Figure 3.1).

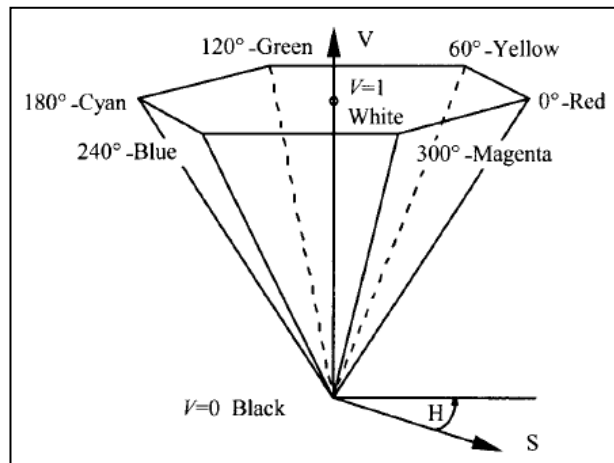


Figure 3.1 Hexacone representation of the HSV color space (Koschan, A. & Abidi, M., 2008).

The saturation S is a number between 0 on the central axis (the V -axis) and 1 on the sides of the pyramid. For $S = 0$, H is undefined. The brightness value V lies between 0 on the apex of the pyramid and 1 on the base. The point on the apex of the

pyramid with $V = 0$ is black. At this point, the values of H and S have no significance. The lightest colors lie on the topside of the pyramid; however, not all colors with the same brightness are visible on the plane $V = 1$.

Figure 3.2 (a) shows a sample RGB image and Figures 3.2 (b) through 3.2 (d) show hue, saturation, and value channels of the HSV image, respectively. The HSV image is created by using `rgb2hsv` command in MATLAB.



Figure 3.2 (a) Original RGB Image (b) Hue channel of the HSV image (c) Saturation channel of the HSV image (d) Value channel of the HSV image.

3.2 Segmentation

Image segmentation is a broad and active field, not only in medical imaging, but also in computer vision and satellite imagery. Its purpose is to divide an image into regions which are meaningful for a particular task. Segmentation is an essential step prior to the description, recognition, or classification of an image or its constituents. There are two major approaches – region-based methods, in which similarities are detected, and boundary-based methods, in which discontinuities (edges) are detected and linked to form boundaries around regions.

Segmentation is the partitioning of an image into meaningful regions, most frequently to distinguish objects or regions of interest (foreground) from everything else (background). In the simplest cases, there would be only these two classes (foreground and background) and the segmented image would be a binary image. Segmentation is used, for example: for the detection of organs, such as the brain, heart, lungs, or liver in CT or MR images; to distinguish pathological tissue, such as a tumor, from normal tissue; and in treatment planning. Pseudocolor can be added to the original image based on the extent of the segmented regions (Figure 3.3). The most basic attribute used in defining the regions is image gray level or brightness, but other properties such as color or texture can also be used. Segmentation is often the first stage in pattern recognition systems; once the objects of interest are isolated from the rest of the image, certain characterizing measurements could be made (feature extraction), and this could be used to classify the objects into particular groups or classes.

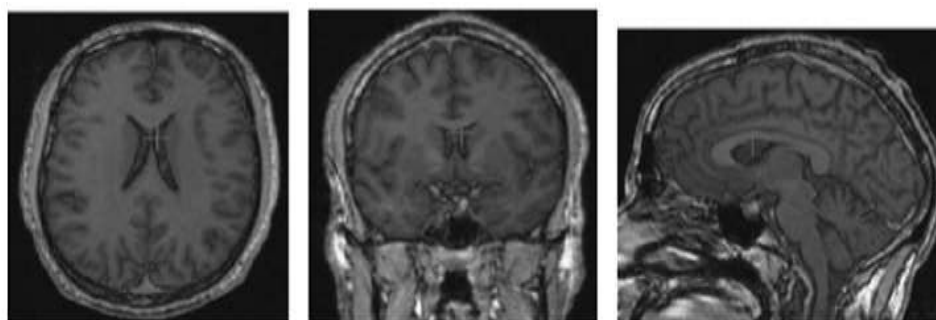


Figure 3.3 A characteristic shading has been added to the brain following segmentation (Dougherty, G., 2009).

3.3 Morphological Operators

Morphological image processing is a tool for extracting or modifying information on the shape and structure of objects within an image. Morphological operators, such as dilation, erosion, and skeletonization are particularly useful for the analysis of binary images, although they can be extended for use with grayscale images as well. Morphological operators are non-linear. Their common usages include filtering, edge detection, feature detection, counting objects in an image, image segmentation, noise reduction, and finding the midline of an object.

There are a number of morphological operators, but the two most fundamental operations are dilation and erosion; all other morphological operations are built from a combination of these two.

3.3.1 Dilation and Erosion

In binary images dilation is an operation that increases the size of foreground objects, generally taken as white pixels, although in some implementations this convention is reversed. Figure 3.4 represents the dilation operation.



Figure 3.4 Dilation operation, (a) Original image, (b) After a single dilation, (c) After several dilations (Dougherty, G., 2009).

Erosion is an operation that increases the size of background objects (and shrinks the foreground objects) in binary images. Figure 3.5 represents the erosion operation.



Figure 3.5 Erosion operation, (a) Original image, (b) After single erosion, (c) After two erosions (Dougherty, G., 2009).

3.3.2 Opening and Closing

The combination of an erosion followed by a dilation is called an opening, referring to the ability of this combination to open up gaps between just-touching features, as shown in Figure 3.6. It is one of the most commonly used sequences for removing fine lines and isolated pixel noise from binary images. Performing the same operations in the opposite order (dilation followed by erosion) produces a different result. This sequence is called a closing because it can close breaks in features (Figure 3.6).

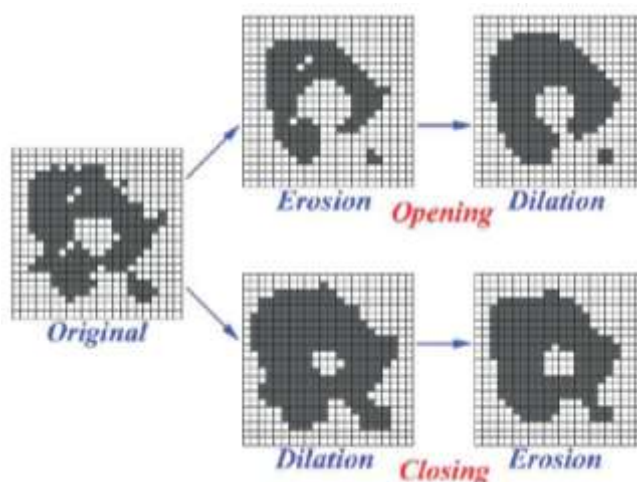


Figure 3.6 Combining erosion and dilation to produce an opening or a closing (Russ, J. C., 2011).

3.4 Feature Extraction

In pattern recognition and image processing, feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and is suspected to be notoriously redundant, then the input data is transformed into a reduced representation set of features. Transforming the input data into the reduced set of features is called feature extraction. If the features extracted are carefully chosen, it is expected that the feature set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the full size input.

3.4.1 Low Level Feature Extraction

Low-level features are those basic features that can be extracted automatically from an image without any shape information. The low-level features are edge detection, corner detection, motion detection, detecting image curvature, etc. as seen in Figure 3.7.

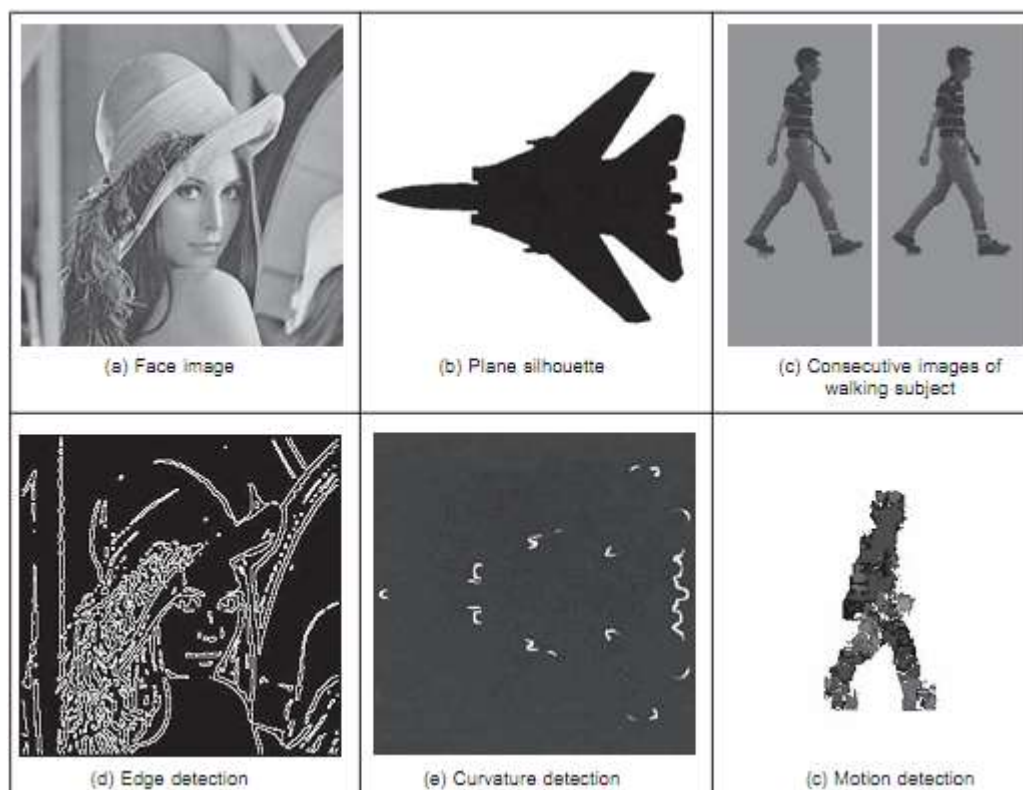


Figure 3.7 Low-level feature extraction (Nixon, M. S. & Aguado, A. S., 2002).

3.4.2 Object Description

Objects are represented as a collection of pixels in an image. Thus, for purposes of recognition we need to describe the properties of groups of pixels. The description is often just a set of numbers – the object's descriptors. Using these, objects by simply matching the descriptors of objects in an image against the descriptors of known objects can be compared and recognized. However, in order to be useful for recognition, descriptors should have four important properties. First, they should define a complete set. That is, two objects must have the same descriptors if and only if they have the same shape. Secondly, they should be congruent. As such, we should

be able to recognize similar objects when they have similar descriptors. Thirdly, it is convenient that they have invariance properties. For example, rotation invariant descriptors will be useful for recognizing objects whatever their orientation. Other important invariance properties naturally include scale, position, and also invariance to affine and perspective changes. Table 3.1 presents the characterization of objects by two forms of descriptors (Nixon, M. S. & Aguado, A. S., 2002).

Table 3.1 Object descriptors (Nixon, M. S. & Aguado, A. S., 2002).

Object Description	Shape Boundary	Chain Codes	
		Fourier Descriptors	Cumulative Angular Function
			Elliptic descriptors
	Region	Basic	Area
			Perimeter
			Compactness
			Dispersion
		Moments	First order
			Centralized
			Zernike

3.4.3 Texture Description

Texture is actually a very nebulous concept, often attributed to human perception, as either the feel or the appearance of (woven) fabric. Everyone has their own interpretation as to the nature of texture; there is no mathematical definition for texture, it simply exists.

As an alternative definition of texture, it can be considered as a database of images that researchers use to test their algorithms. Many texture researchers have used a database of pictures of textures, produced for artists and designers, rather than for digital image analysis. Parts of three of the Brodatz texture images are given in Figure 3.8.

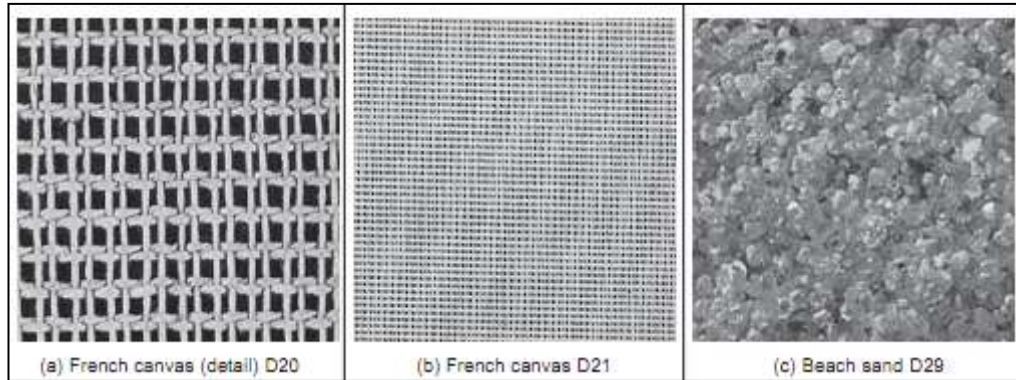


Figure 3.8 Three Brodatz textures (Nixon M. S. & Aguado A. S., 2002).

Clearly, images will usually contain samples of more than one texture. Accordingly, we would like to be able to describe texture (texture descriptions are measurements which characterize a texture) and then to classify it (classification is attributing the correct class label to a set of measurements) and then perhaps to segment an image according to its textural content. Basically, there are three approaches for deriving the features which can be used to describe textures. These can be given as follows;

- Structural (transform-based) approaches
- Statistical approaches
- Combination approaches

3.4.3.1 Statistical Approaches

The most famous statistical approach is the co-occurrence matrix. This was the result of the first approach to describe, and then classify, image texture (Haralick, R. M., 1979). It remains popular today, by virtue of good performance. The co-occurrence matrix contains elements that are counts of the number of pixel pairs for specific brightness levels, when separated by some distance and at some relative inclination. For brightness levels $b1$ and $b2$ the co-occurrence matrix C is

$$C_{b1,b2} = \sum_{x=1}^N \sum_{y=1}^N (P_{x,y} = b1) \wedge (P_{x',y'} = b2) \quad (3.1)$$

where the x co-ordinate x' is the offset given by the specified distance d and inclination θ by

$$x' = x + d \cos(\theta) \quad \forall d \in [1, \max(d)] \wedge \theta \in [0, 2\pi], \quad (3.2)$$

and the y co-ordinate y' is

$$y' = y + d \sin(\theta) \quad \forall d \in [1, \max(d)] \wedge \theta \in [0, 2\pi]. \quad (3.3)$$

When Equation 3.1 is applied to an image, we obtain a square and symmetric matrix whose dimensions equal the number of grey levels in the picture. The co-occurrence matrices for the three Brodatz textures of Figure 3.8 are shown in Figure 3.9. The results for the two samples of French canvas in Figures 3.9 (a) and (b) appear to be much more similar and quite different than the co-occurrence matrix for beach sand in Figure 3.9 (c). As such, the co-occurrence matrix looks like it can better expose the underlying nature of texture than can the Fourier description. This is because co-occurrence measures spatial relationships between brightness, as opposed to frequency content.

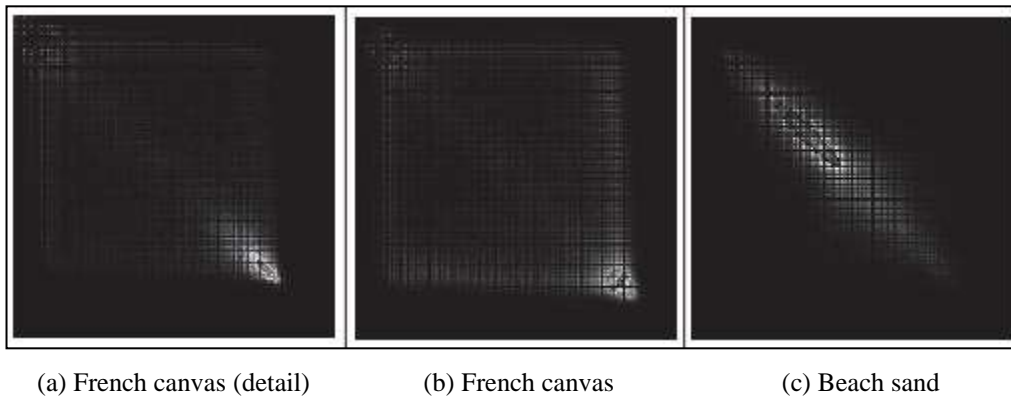


Figure 3.9 Co-occurrence matrices of the three Brodatz textures (Nixon M. S. & Aguado A. S., 2002).

Haralick described 14 statistics that can be calculated from the co-occurrence matrix with the intent of describing the texture of the image. Table 3.2 shows Haralick's features for textural analysis.

Table 3.2 Features in Haralick's co-occurrence-based method for texture analysis (Boland M. V., 1999).

Angular Second Moment	$\sum_i \sum_j p(i, j)^2$
Contrast	$\sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g} \sum_{j=1}^{N_g} p(i, j) \right\}, i - j = n$
Correlation	$\frac{\sum_i \sum_j (ij)p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y}$ where μ_x , μ_y , σ_x , and σ_y are the means and std. deviations of p_x and p_y , the partial probability density functions
Sum of Squares: Variance	$\sum_i \sum_j (1 - \mu)^2 p(i, j)$
Inverse Difference Moment	$\sum_i \sum_j \frac{1}{1+(i-j)^2} p(i, j)$
Sum Average	$\sum_{i=2}^{2N_g} i p_{x+y}(i)$ where x and y are the coordinates (row and column of an entry in the co-occurrence matrix and $p_{x+y}(i)$ is the probability of the co-occurrence matrix coordinates summing to $x+y$
Sum Variance	$\sum_{i=2}^{2N_g} (i - f_s)^2 p_{x+y}(i)$
Sum Entropy	$-\sum_{i=2}^{2N_g} p_{x+y}(i) \log\{p_{x+y}(i)\} = f_s$
Entropy	$-\sum_i \sum_j p(i, j) \log(p(i, j))$
Difference Variance	$\sum_{i=0}^{N_g-1} i^2 p_{x-y}(i)$
Difference Entropy	$-\sum_{i=0}^{N_g-1} p_{x-y}(i) \log\{p_{x-y}(i)\}$
Info. Measure of Correlation 1	$\frac{HXY - HXY1}{\max\{HX, HY\}}$
Info. Measure of Correlation 2	$(1 - \exp[-2(HXY2 - HXY)])^{\frac{1}{2}}$ where $HXY = -\sum_i \sum_j p(i, j) \log(p(i, j))$, HX, HY are the entropies of p_x and p_y , $HXY1 = -\sum_i \sum_j p(i, j) \log\{p_x(i)p_y(j)\}$ $HXY2 = -\sum_i \sum_j p_x(i)p_y(j) \log\{p_x(i)p_y(j)\}$
Max. Correlation Coefficient	Square root of the second largest eigenvalue of Q where $Q(i, j) = \sum_k \frac{p(i, k)p(j, k)}{p_x(i)p_y(j)}$

3.4.4 Dimensionality Reduction

In machine learning, dimension reduction is the process of reducing the number of variables under consideration. Dimensionality reduction techniques can be divided into two groups as feature selection and feature extraction.

3.4.4.1 Principal Component Analysis (PCA)

Component analysis is the most popular feature extraction technique, especially PCA. Component analysis is an unsupervised approach to finding the “right” features from the data. We discuss two leading methods (PCA and ICA), each having a somewhat different goal.

In PCA, we seek to represent the d -dimensional data in a lower-dimensional space. This will reduce the degrees of freedom, reduce the space and time complexities. The goal is to represent data in a space that best describes the variation in a sum-squared error sense. In independent component analysis (ICA), we seek those directions that show the independence of signals. This method is particularly helpful for separating signals from multiple sources. As with standard clustering methods, it helps greatly if we know how many independent components exist ahead of time (Duda, R. O., Hart, P. E. & Stork, D. G., 2000).

3.4.4.2 Feature Selection

Feature selection is the technique of selecting a subset of relevant features for building robust learning models. Feature selection is a particularly important step in analyzing the data from many experimental techniques in biology, such as DNA microarrays, because they often entail a large number of measured variables (features) but a very low number of samples. By removing most irrelevant and redundant features from the data, feature selection helps improve the performance of learning models by:

- Alleviating the effect of the curse of dimensionality
- Enhancing generalization capability
- Speeding up learning process
- Improving model interpretability.

Feature selection also helps people acquire better understanding about their data by telling them which are the important features and how they are related with each other.

Simple feature selection algorithms are ad hoc, but there are also more methodical approaches. From a theoretical perspective, it can be shown that optimal feature selection for supervised learning problems requires an exhaustive search of all possible subsets of features of the chosen cardinality. If large numbers of features are available, this is impractical. For practical supervised learning algorithms, the search is for a satisfactory set of features instead of an optimal set.

Feature selection algorithms typically fall into two categories: feature ranking and subset selection. Feature ranking ranks the features by a metric and eliminates all features that do not achieve an adequate score. Subset selection searches the set of possible features for the optimal subset.

In statistics, the most popular form of feature selection is stepwise regression. It is a greedy algorithm that adds the best feature (or deletes the worst feature) at each round. The main control issue is deciding when to stop the algorithm. In machine learning, this is typically done by cross-validation. In statistics, some criteria are optimized. This leads to the inherent problem of nesting. More robust methods have been explored, such as branch and bound and piecewise linear network.

CHAPTER FOUR

THEORETICAL BACKGROUND ON ARTIFICIAL NEURAL NETWORKS

4.1 Computational Models of Neurons

McCulloch and Pitts proposed a binary threshold unit as a computational model for a neuron. A schematic diagram of a McCulloch Pitts neuron is shown in Figure 4.1 (Jain, A. K., Mao, J. & Mohiuddin, K. M., 1996).

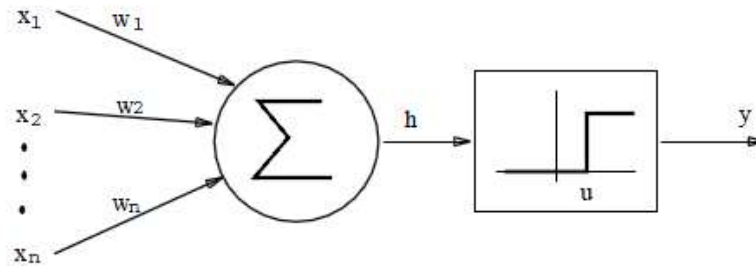


Figure 4.1 McCulloch-Pitts model of a neuron.

This mathematical neuron computes a weighted sum of its n input signals, $x_j, j = 1, 2, \dots, n$, and generates an output of 1 if this sum is above a certain threshold u , and an output of 0, otherwise. Mathematically,

$$y = \theta(\sum_{j=1}^n w_j x_j - u), \quad (4.1)$$

where $\theta(\cdot)$ is a unit step function, and w_j is the synapse weight associated with the j^{th} input. For simplicity of notation, we often consider the threshold u as another weight $w_0 = -u$ attached to the neuron with a constant input $x_0 = 1$. Positive weights correspond to *excitatory* synapses, while negative weights model *inhibitory* ones.

The McCulloch-Pitts neuron has been generalized in many ways. An obvious one is to use activation functions other than the threshold function, such as piecewise linear, sigmoid, or Gaussian, as shown in Figure 4.2. The sigmoid function is by far the most frequently used in artificial neural networks (ANNs). The standard sigmoid function is the *logistic* function, defined by

$$g(x) = \frac{1}{1 + \exp(-\beta x)} \quad (4.2)$$

where β is the slope parameter.

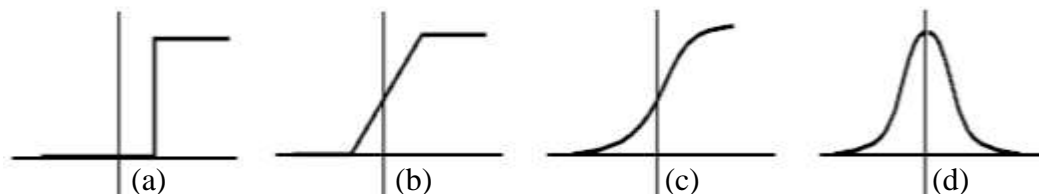


Figure 4.2 Different types of activation functions; (a) Threshold, (b) Piecewise linear, (c) Sigmoid, (d) Gaussian (Jain, A. K., Mao, J. & Mohiuddin, K. M., 1996).

4.2 Network Architectures

ANNs can be viewed as weighted directed graphs in which artificial neurons are nodes and directed edges (with weights) are connections between neuron outputs and neuron inputs. Based on the connection pattern (architecture), ANNs can be grouped into two categories (Figure 4.3):

- *feed-forward* networks, in which graphs have no loops
- *recurrent* (or *feedback*) networks, in which loops occur because of feedback connections

In the most common family of feed-forward networks, called multilayer perceptron, neurons are organized into layers that have unidirectional connections between them. Figure 4.3 also shows typical networks for each category.

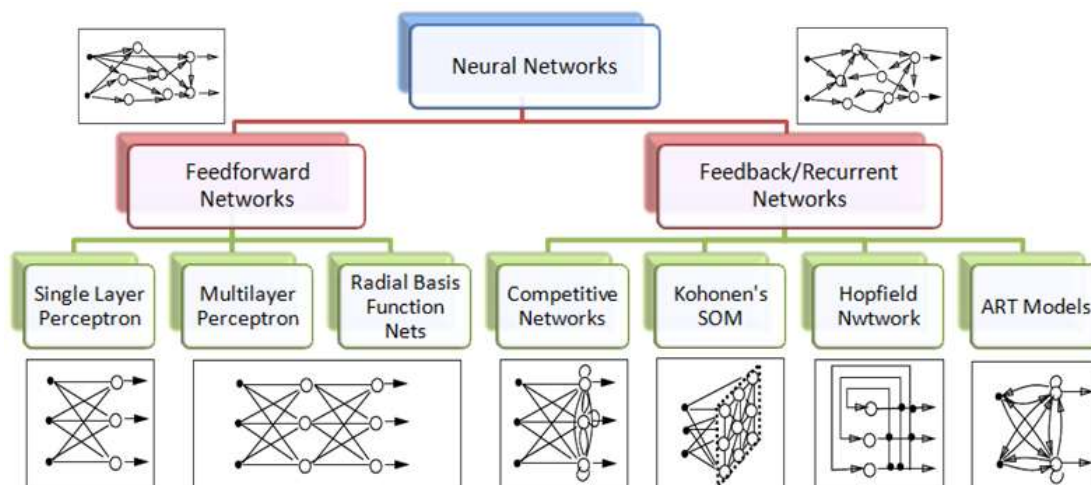


Figure 4.3 A taxonomy of network architectures (Jain, A. K., Mao, J. & Mohiuddin, K. M., 1996).

Different connectivities yield different network behaviors. Generally speaking, feed-forward networks are *static*, that is, they produce only one set of output values rather than a sequence of values from a given input. Feed-forward networks are memoryless in the sense that their response to an input is independent of the previous network state. Recurrent, or feedback, networks, on the other hand, are dynamic systems. When a new input pattern is presented, the neuron outputs are computed.

4.3 Learning

Ability to learn is a fundamental trait of intelligence. Although what is meant by learning is often difficult to describe, a learning process, in the context of artificial neural networks can be viewed as the problem of updating network architecture and connection weights so that a network can efficiently perform a specific task. Typically, learning in ANNs is performed in two ways. Sometimes, weights can be set a priori by the network designer through a proper formulation of the problem. However, most of the time, the network must learn the connection weights from the given training patterns. Improvement in performance is achieved over time through iteratively updating the weights in the network.

There are three main learning paradigms: supervised, unsupervised, and hybrid. In supervised learning, or learning with a teacher, the network is provided with a correct

answer (output) for every input pattern. Weights are determined to allow the network to produce answers as close as possible to the known correct answers. In contrast, unsupervised learning, or learning without a teacher, does not require a correct answer associated with each input pattern in the training data set. Hybrid learning combines supervised and unsupervised learning. Part of the weights is usually determined through supervised learning, while the others are obtained through unsupervised learning.

Learning theory must address three fundamental and practical issues associated with learning from samples: capacity, sample complexity, and time complexity. There are four basic types of learning rules: error correction, Boltzmann, Hebbian, and competitive learning (Figure 4.4) (Jain, A. K., Mao, J. & Mohiuddin, K. M., 1996).

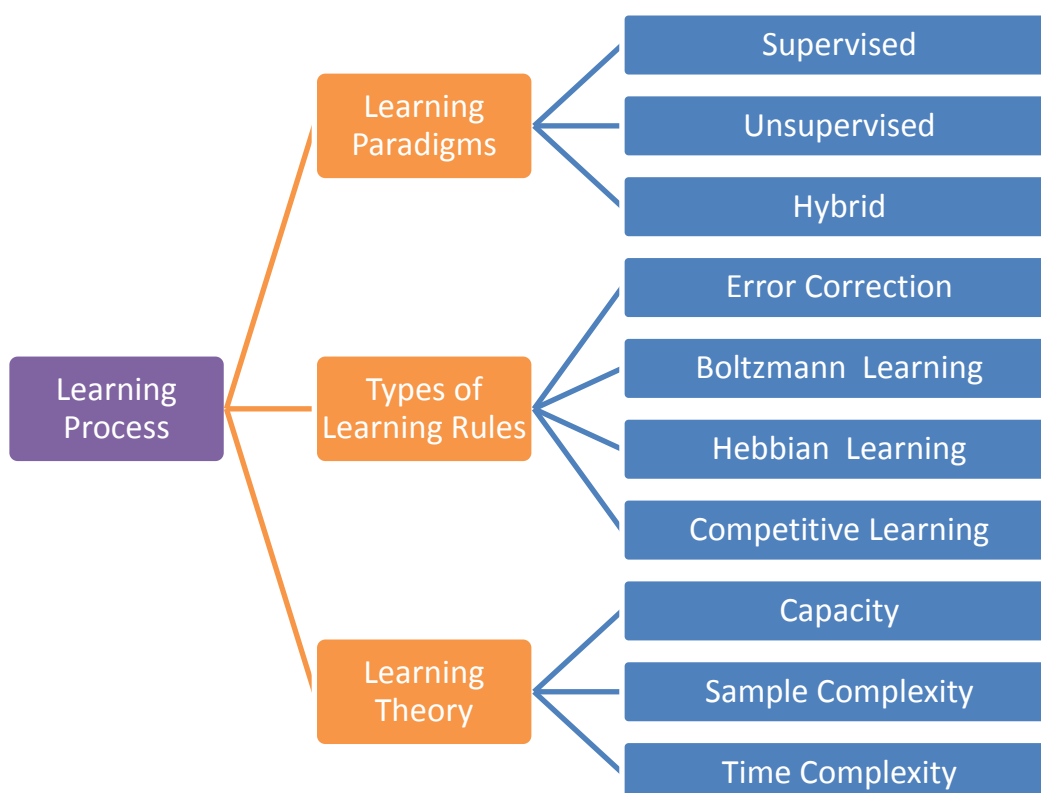


Figure 4.4 Elements of the learning process (Jain, A. K., Mao, J. & Mohiuddin, K. M., 1996).

4.4 Resampling Techniques

In statistics, resampling is any of a variety of methods for doing one of the following:

- Estimating the precision of sample statistics (medians, variances, percentiles) by using subsets of available data (jackknifing) or drawing randomly with replacement from a set of data points (bootstrapping)
- Exchanging labels on data points when performing significance tests (permutation tests, also called exact tests, randomization tests, or re-randomization tests)
- Validating models by using random subsets (bootstrapping, cross validation)

Common resampling techniques are given in the following sections.

4.4.1 Cross Validation

In cross validation we randomly split the set of labeled training samples D into two parts: one is used as the traditional training set for adjusting model parameters in the classifier. The other set — the validation set — is used to estimate the generalization validation error. Since our ultimate goal is low generalization error, we train the classifier until we reach a minimum of this validation error, as sketched in Figure 4.5. It is essential that the validation set not include points used for training the parameters in the classifier — a methodological error known as “testing on the training set” (Duda, R. O., Hart, P. E. & Stork, D. G., 2000).

4.4.1.1 M -Fold Cross Validation

A simple generalization of the above method is m -fold cross validation. Here the cross validation training set is randomly divided into m disjoint sets of equal size n/m , where n is the total number of patterns in D . The classifier is trained m times, each time with a different set held out as a validation set. The estimated performance is the mean of these m errors. In the limit where $m = n$, the method is in effect the leave-one-out approach.

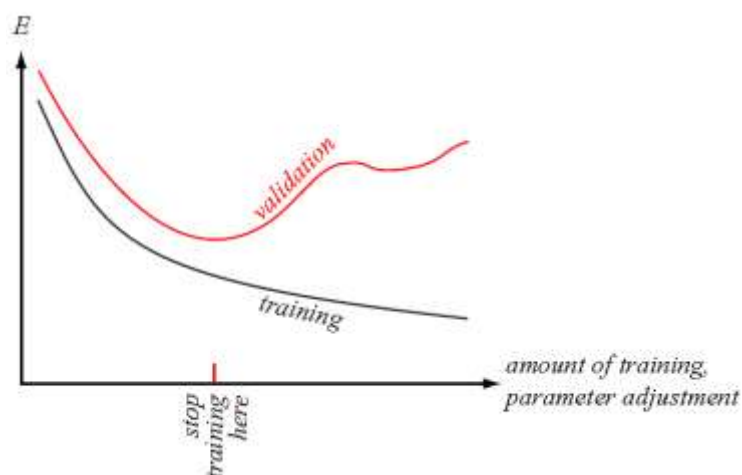


Figure 4.5 Training with cross validation.

4.4.1.2 Leave-One-Out Cross Validation

As the name suggests, leave-one-out cross-validation involves using a single observation from the original sample as the validation data, and the remaining observations as the training data. This is repeated such that each observation in the sample is used once as the validation data. This is the same as a K -fold cross-validation with K being equal to the number of observations in the original sample. Leave-one-out cross-validation is computationally expensive because it requires many repetitions of training.

4.4.2 Bootstrap

A “bootstrap” data set is one created by randomly selecting n points from the training set D , with replacement. In bootstrap estimation, this selection process is independently repeated B times to yield B bootstrap data sets, which are treated as independent sets. There are several ways to generalize the bootstrap method to the problem of estimating the accuracy of a classifier. One of the simplest approaches is to train B classifiers, each with a different bootstrap data set, and test on other bootstrap data sets. The bootstrap estimate of the classifier accuracy is simply the mean of these bootstrap accuracies. In practice, the high computational complexity of bootstrap estimation of classifier accuracy is rarely worth possible improvements in that estimate (Duda, R. O., Hart, P. E. & Stork, D. G., 2000).

4.4.3 Boosting

The goal of boosting is to improve the accuracy of any given learning algorithm. In boosting we first create a classifier with accuracy on the training set greater than average, and then add new component classifiers to form an ensemble whose joint decision rule has arbitrarily high accuracy on the training set. In such a case we say the classification performance has been “boosted.” In overview, the technique trains successive component classifiers with a subset of the training data that is “most informative” given the current set of component classifiers.

4.4.3.1 AdaBoost

There are a number of variations on basic boosting. The most popular, AdaBoost — from “adaptive” boosting — allows the designer to continue adding weak learners until some desired low training error has been achieved. In AdaBoost, each training pattern receives a weight which determines its probability of being selected for a training set for an individual component classifier. If a training pattern is accurately classified, then its chance of being used again in a subsequent component classifier is reduced; conversely, if the pattern is not accurately classified, then its chance of being used again is raised. In this way, AdaBoost “focuses on” the informative or “difficult” patterns.

AdaBoost applied to a weak learning system can reduce the training error E exponentially as the number of component classifiers, k_{max} , is increased. Because AdaBoost “focuses on” difficult training patterns, the training error of each successive component classifier (measured on its own weighted training set) is generally larger than that of any previous component classifier (shown in gray in Figure 4.6). It is often found that the train and test errors decrease in boosted systems as well, as shown in red in Figure 4.6.



Figure 4.6 Ensemble train and test error graph (Duda, R. O., Hart, P. E. & Stork, D. G., 2000).

CHAPTER FIVE THE EXPERIMENTAL STUDY

In this chapter, details of the MATLAB algorithms developed in this thesis are given step by step.

As mentioned before, the aim of this thesis is to determine the stage of the NB disease by examining the neuroblastoma tissue images. For this purpose, the neuropil percentage, the number of mitosis and karyorrhexis cells, and the number of differentiated cells are estimated by our algorithms.

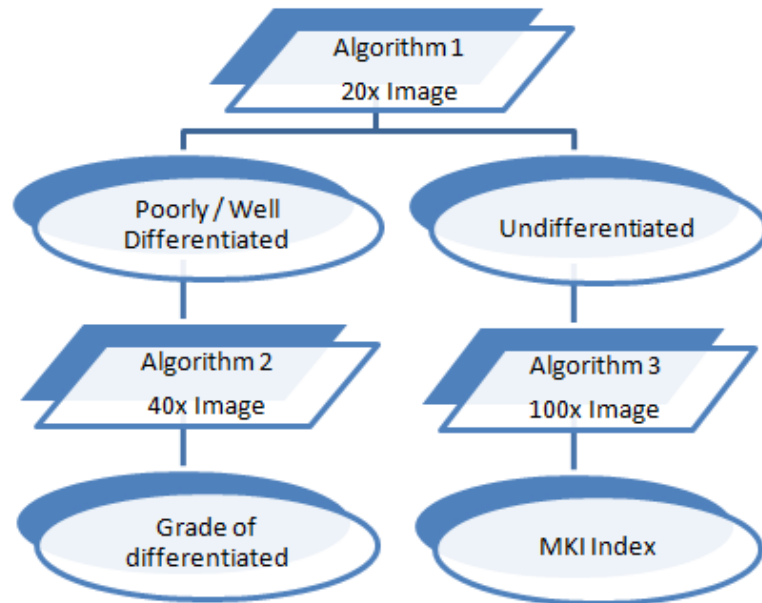


Figure 5.1 The relationship among various algorithms and the use of images with different magnifications.

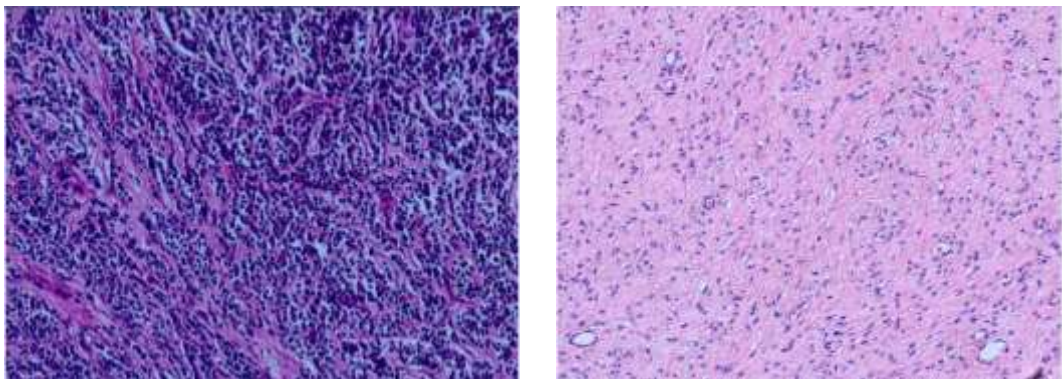


Figure 5.2 Tissue images of NB tumor with 20x zoom.

There are three different algorithms developed in this study as seen in Figure 5.1. The first algorithm is used for determining the neuropil percentage of the 20x image (Figure 5.2). We have named this algorithm “Neuropil Percentage Determination Algorithm (NPDA)”. If the output of NPDA comes up as poorly differentiated or well differentiated, the second algorithm must be used and the input image taken under 40x magnification from the same region (Figure 5.3 (a) and (b)) must be processed. We have named the second algorithm “Cell Detection in Differentiated Images (CDDI)”. The aim of the CDDI algorithm is to create a binary image belonging to differentiated cells in the differentiated image and the other tumor cells. On the other hand, if the output of NPDA is determined as undifferentiated, the third algorithm, which we call “Cell Detection in Undifferentiated Images (CDUI)”, must be used and the input image taken under 100x magnification from the same region (Figure 5.3 (c) and (d)) must be processed. The aim of CDUI algorithm is to create a binary image belonging to mitosis and karyorrhexis cells in the undifferentiated image and the other tumor cells.

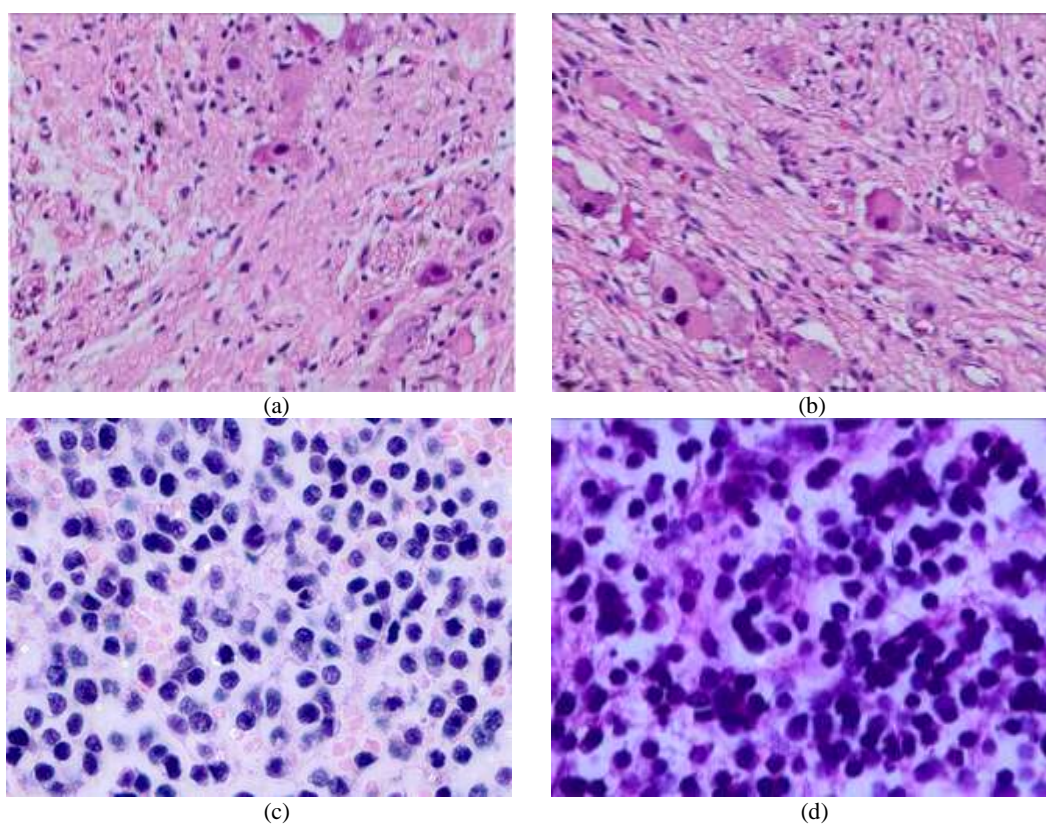


Figure 5.3 (a), (b) Tissue images of NB tumor including differentiated cells with 40x zoom, (c), (d) Tissue images of NB tumor including mitosis and karyorrhexis cells with 100x zoom.

5.1 Detection of Neuropil Regions

First of all, the percentage of the neuropil structure is determined in the images which are taken under 20x magnification to decide if the image includes mitosis and karyorrhexis cells or differentiated cells. If the sample images in Figure 5.4 are examined, it is seen that nuclei, cytoplasm, neuropil, and the other components can be segmented by using the color information. In the images in Figure 5.4, dark purple color corresponds to tumor nuclei, and light pink color indicates neuropil. If light pink color is detected and the other color components are not taken into account, the neuropil can be segmented.

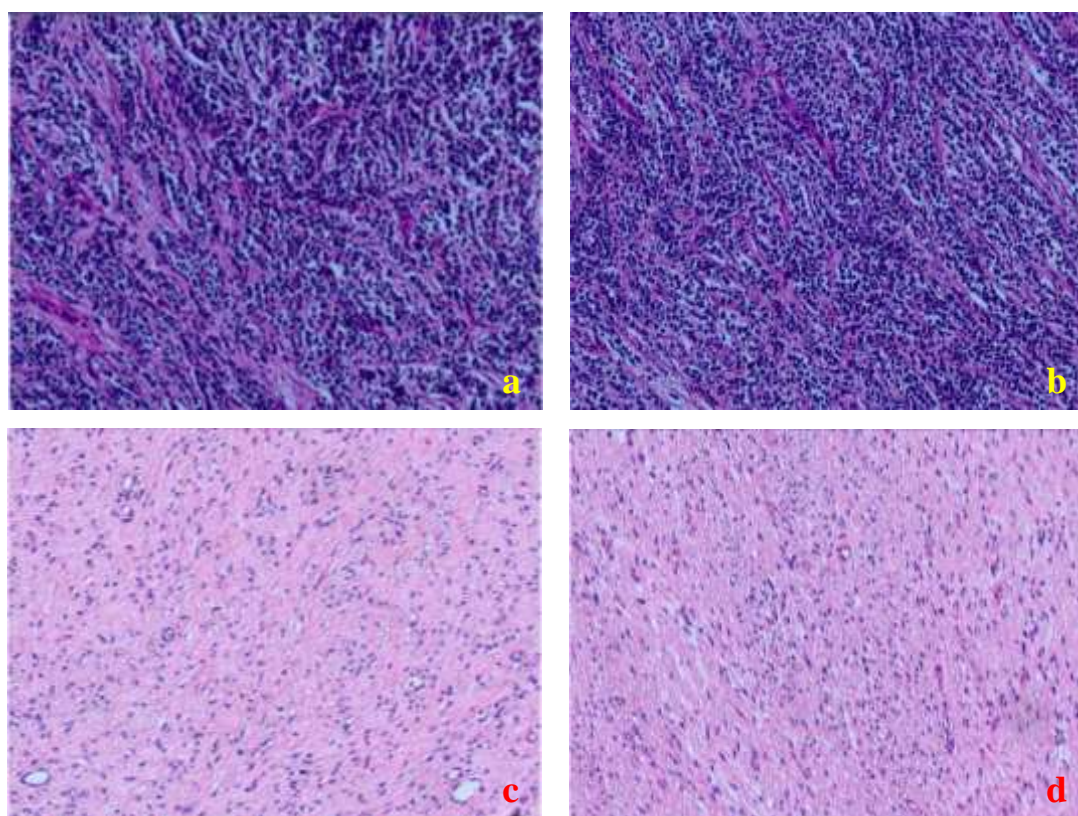


Figure 5.4 (a), (b) Tissue images of NB tumor of undifferentiated type with 20x zoom, (c), (d) Tissue images of NB tumor of differentiated type with 20x zoom.

In order to determine the algorithm to be used in the next stage, the percentage of neuropil must be determined. If the image is decided to have neuropil regions covering higher than 5% of the whole image, then the image type is determined as differentiated (poorly differentiated or well differentiated). If the image is decided to

have neuropil regions covering less than 5%, it means the image type is undifferentiated.

The flowchart of NPDA is given in Figure 5.5. This algorithm is used to determine the neuropil percentage in the image which is taken under 20x magnification.

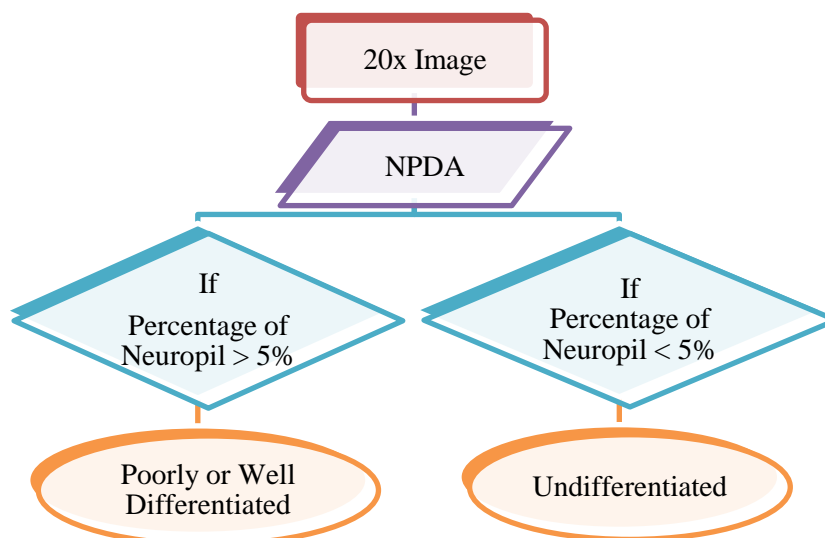


Figure 5.5 The flowchart of NPDA.

The color image belonging to neuroblastoma tissue (Figure 5.6 (a)) is obtained by taking the image of real neuroblastoma tissue samples by using an electron microscope in our laboratory. Hue, saturation, and value channels of the color image are shown in Figure 5.6 (b), (c), and (d), respectively.

After the pixel values in the value channel are set to one, the color image given in Figure 5.7 is obtained. While constructing the binary mask, the value channel is not used. Since the value channel is related with only the brightness of the image, it does not affect the color information which we want to detect (Figure 5.8).

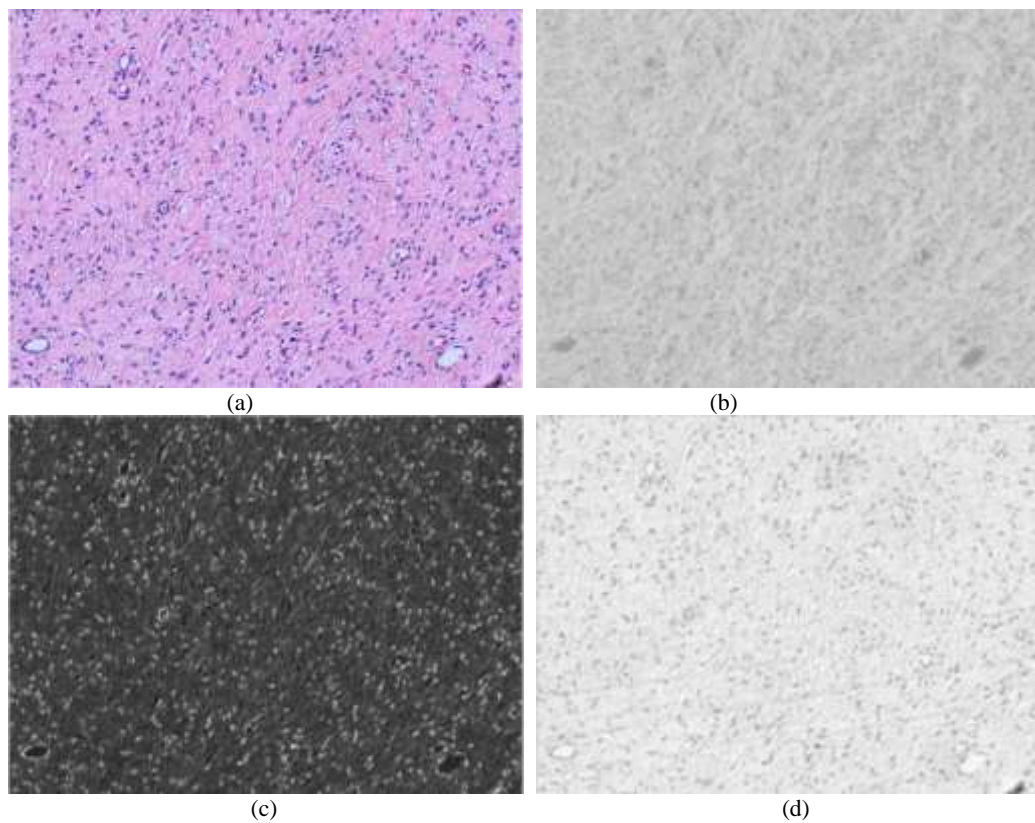


Figure 5.6 (a) Tissue image of NB tumor with 20x zoom, (b) Hue channel, (c) Saturation channel, (d) Value channel.

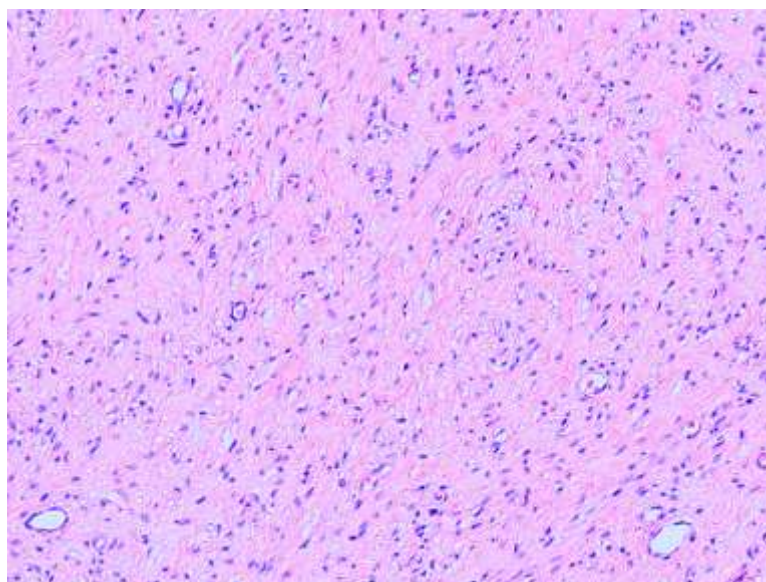


Figure 5.7 The obtained color image when the value channel of the image is set to one.

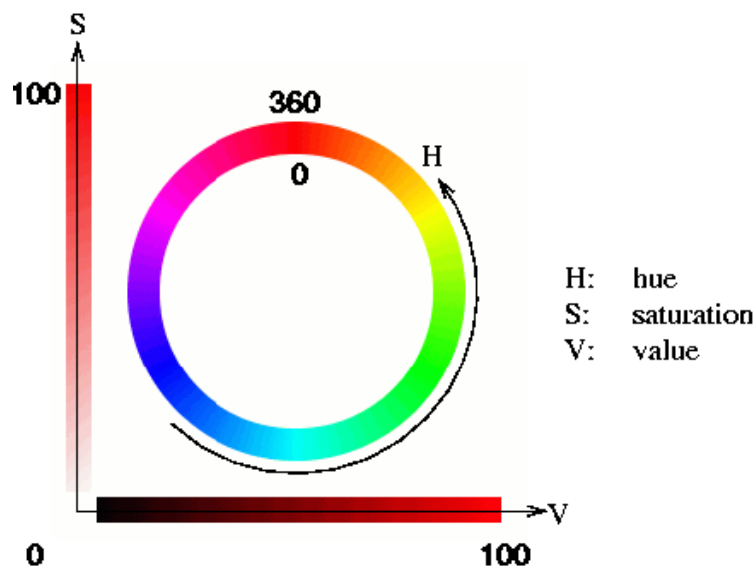


Figure 5.8 HSV cylinder and hue, saturation, and brightness scale (<http://ie.technion.ac.il/CC/Gimp/node51.html>).

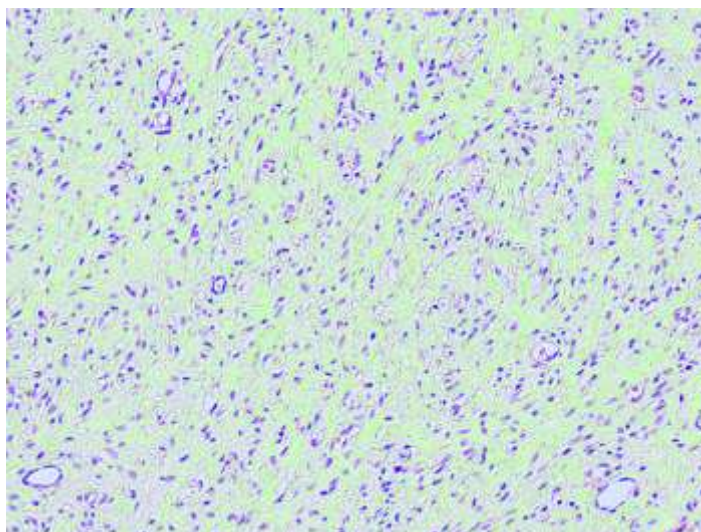


Figure 5.9 The labeled neuropil regions (with green color).

By using the color information, the binary image to be used for masking out the neuropil regions is created. For this purpose, thresholding is applied to hue and saturation channels. By this way, the selected color is converted into white and the other colors are converted into black and the final binary image is obtained. We have named this algorithm “Color Detection Algorithm (CODA)”. The inputs of this algorithm are hue and saturation channels of the color image, minimum hue threshold level, maximum hue threshold level, and saturation threshold level. The output of the algorithm is a binary image which has white-colored regions for the

detected color and black-colored regions for the eliminated colors. If the pixel value of the hue channel is between minimum and maximum hue threshold levels and also the pixel value of the saturation channel is higher than the saturation threshold level, the related pixel value is converted into 1; if not, it is converted into 0. This way, the final binary image is obtained.

Most of the neuropil regions seen in Figure 5.9 where the labeling color is green are successfully detected. Figure 5.10 shows the detected neuropil regions by using the binary masking image.

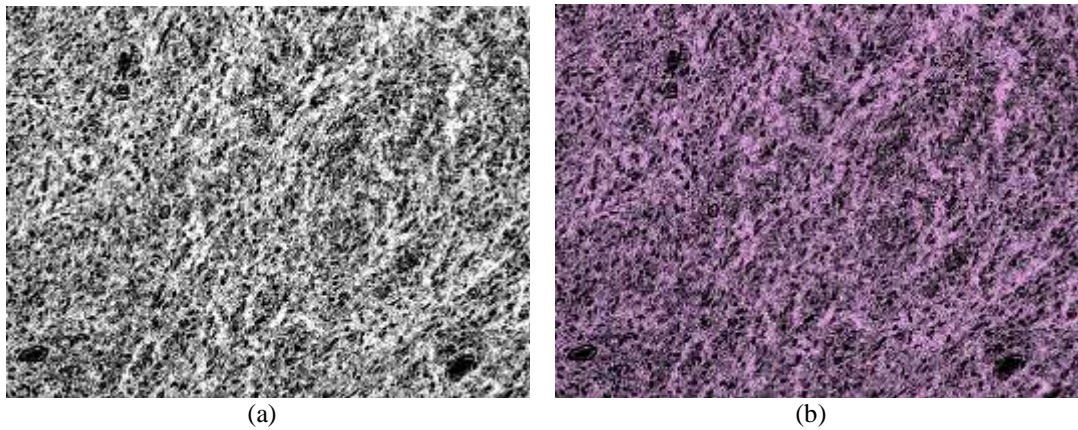


Figure 5.10 (a) The obtained binary masking image for detecting neuropil regions, (b) the image obtained after applying binary AND operation to the mask and the original color image.

CODA algorithm used for detecting neuropil structure can also be used for segmentation of undesired regions. However, the threshold levels that are used at hue and saturation channels are different for that task. Figure 5.11 shows the undesired areas in the image that are detected in order to calculate percentage of neuropil regions.

The percentage of neuropil is calculated as in Equation 5.1.

$$\text{Percentage of Neuropil} = \frac{(\text{Area of Neuropil Regions}) \times 100}{(\text{Image Area}) - (\text{Area of Unwanted Regions})}. \quad (5.1)$$

The percentage of neuropil is 42.22% in the image seen in Figure 5.6 (a). For that reason, the image type is decided as well differentiated, so that a new image must be

taken by using 40x magnification and the CDDI algorithm must be used for analyzing that image.

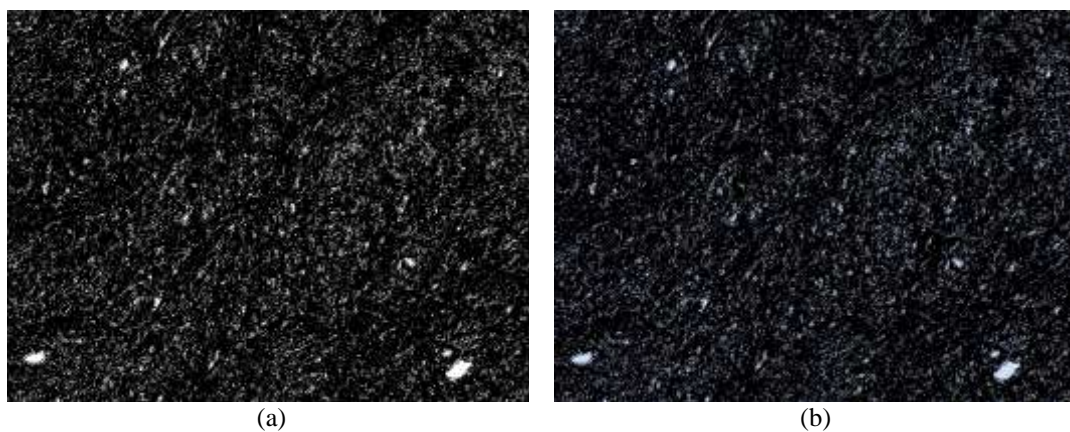


Figure 5.11 (a) The obtained binary masking image for unwanted regions, (b) the image obtained after applying binary AND operation to the mask and the original color image.

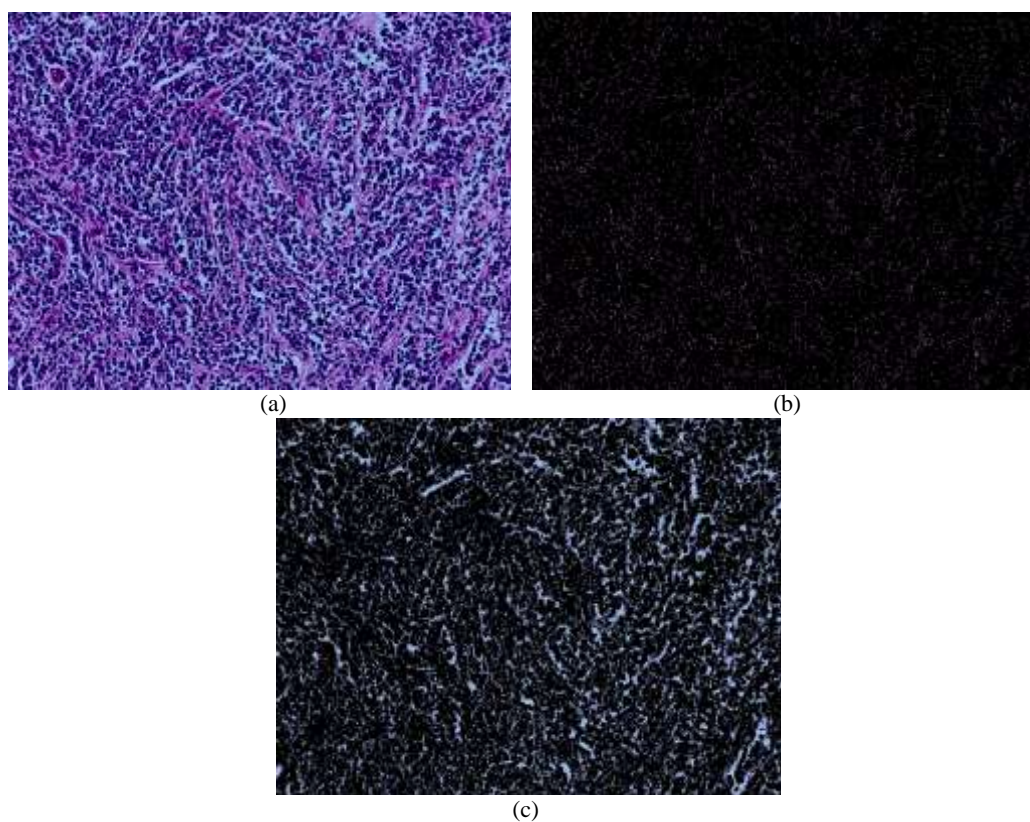


Figure 5.12 (a) Tissue image of NB tumor (undifferentiated type) with 20x magnification, (b) Detected neuropil regions, (c) Detected undesired regions.

The percentage of neuropil in the image seen in Figure 5.12 (a) is found as 2.99% by using NPDA. Thus, the image type is determined as undifferentiated, so that a new image must be taken by using 100x magnification and CDUI algorithm must be used for analyzing that new image. The detected neuropil regions and the detected undesired regions are seen in Figure 5.12 (b) and (c), respectively.

5.2 Detection of Cells in Differentiated Images

If 20x zoomed image has neuropil regions, the tissue will be poorly differentiated or well differentiated, so that a new image which is taken under 40x magnification must be used as input of the CDDI algorithm to create a binary image in order to find the grade of differentiation. Figure 5.13 shows the flowchart of the CDDI algorithm.

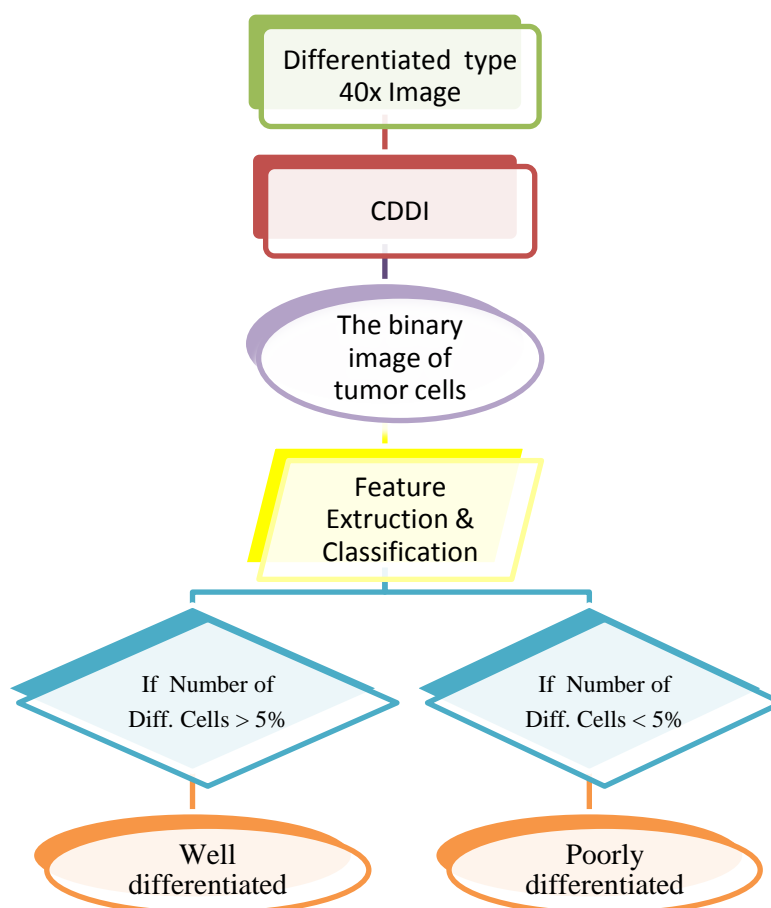


Figure 5.13 The flowchart of the CDDI algorithm.

When the original image in Figure 5.14 is examined, it is seen that there are two different tumor cells; differentiated cells and the other tumor cells. The differentiated cells have cytoplasm, but the other tumor cells do not have cytoplasm. Their color is dark purple and their area is considerably smaller as compared to the differentiated cells. Figure 5.14 also indicates that the color distribution range is rather narrow. In order to separate the differentiated cells and the other tumor cells, firstly the image contrast is enhanced as seen in Figure 5.15. Now, the cells are clearer and most of the background components are eliminated. The image whose contrast is enhanced is used as the input image for detection of white-blue undesired regions, differentiated cells, and the other tumor cells.

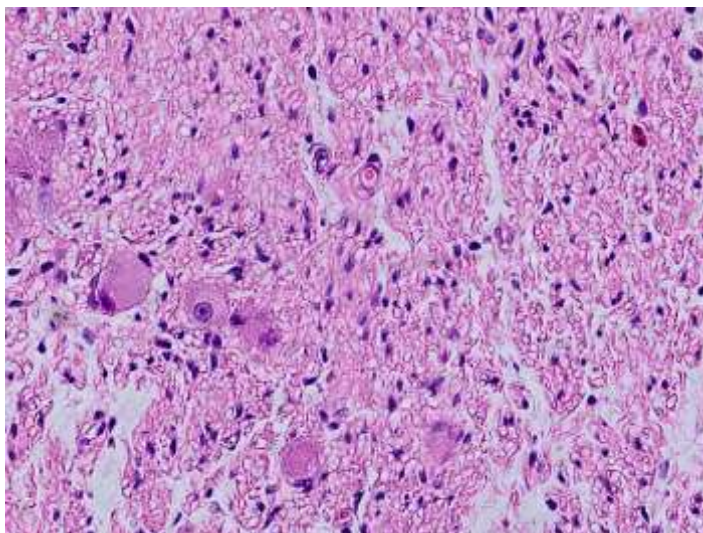


Figure 5.14 The image with 40x magnification.

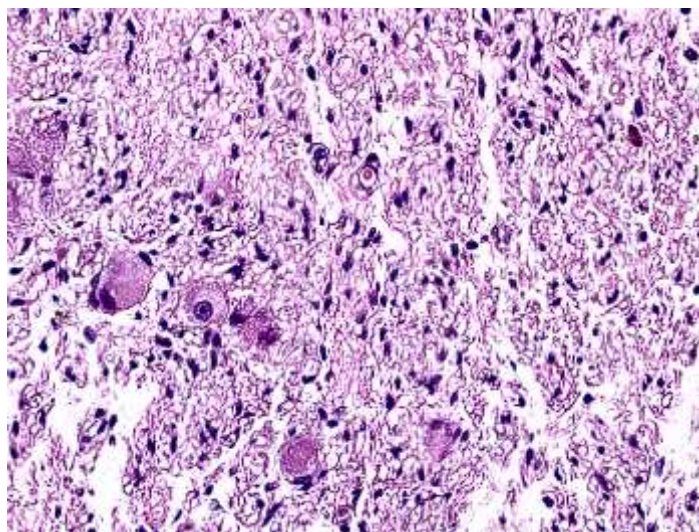


Figure 5.15 The image in Figure 5.14 after contrast enhancement.

Figure 5.16 shows hue, saturation, and value channels of the image in Figure 5.15 whose contrast is enhanced.

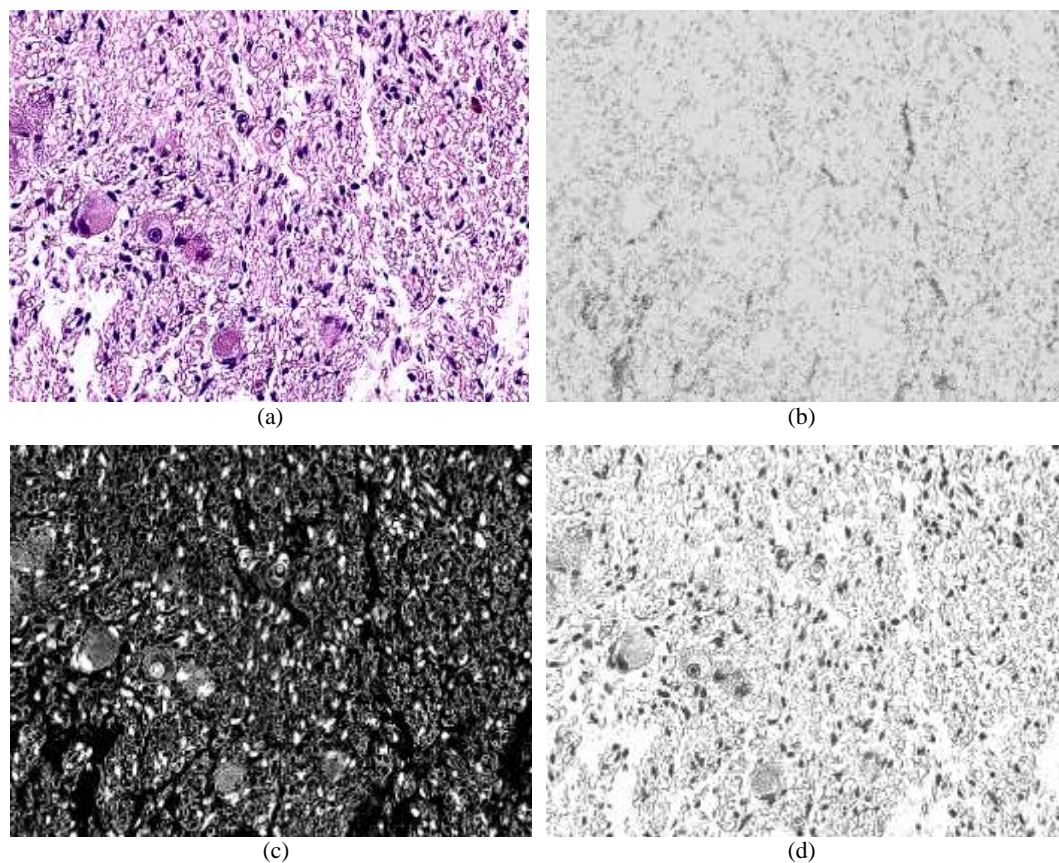


Figure 5.16 (a) Tissue image of NB tumor (differentiated type) with 40x magnification after contrast enhancement, (b) Hue channel, (c) Saturation channel, (d) Value channel.

If we examine the image in Figure 5.16 (a), we see that there are white-blue regions combined with differentiated cells. The white-blue regions must be detected and be subtracted from the image in Figure 5.16 (a) which is the input image of the CDDI algorithm. By utilizing saturation channel of the image (Figure 5.16 (c)), white-blue background is detected using thresholding method as seen in Figure 5.17. By this way, the white-blue background regions can be separated from the differentiated cells.

If the image includes blood cells, these cells can be mistakenly detected as differentiated cells or the other tumor cells. Therefore, they must be detected and eliminated, as well. CODA algorithm is used to detect the blood cells. The minimum

hue threshold level is chosen as 0.85, the maximum hue threshold level is chosen as 1, and the saturation threshold level is chosen as 0.5 to segment blood cells. Figure 5.18 shows some detected blood cells in the image. The regions within the above given threshold values are converted into white and the other regions are converted into black, and thus the binary masking image for blood cells is obtained.

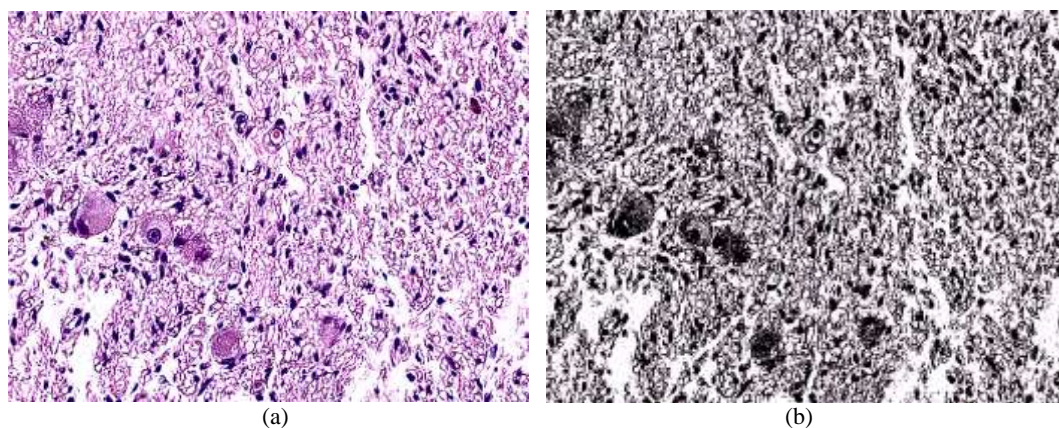


Figure 5.17 (a) The contrast enhanced color image, (b) The detected white-blue regions.

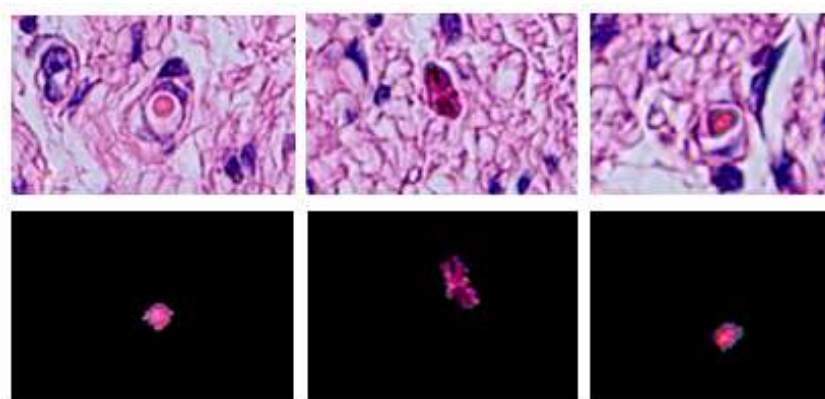


Figure 5.18 Some segmented blood cells.

Up to now, the white-blue regions and blood cells have been detected for the correct segmentation of tumor cells.

The CDDI algorithm enables the differentiated cells and the other tumor cells to be identified in two different ways. In order to detect differentiated cells, the CDDI algorithm uses the saturation channel of the contrast enhanced image in Figure 5.16 (c). Thresholding is applied to saturation channel of the contrast enhanced image. If

the pixel value is larger than 0.3, the pixel value is converted into 1; otherwise it is converted into 0. The obtained binary image is shown in Figure 5.19 (a). The holes are filled, the white-blue regions and blood cells are eliminated, and binary AND operation is applied to the rough binary image and the original tissue image in Figure 5.14 as seen in Figure 5.19 (b), (c) and (d), respectively.

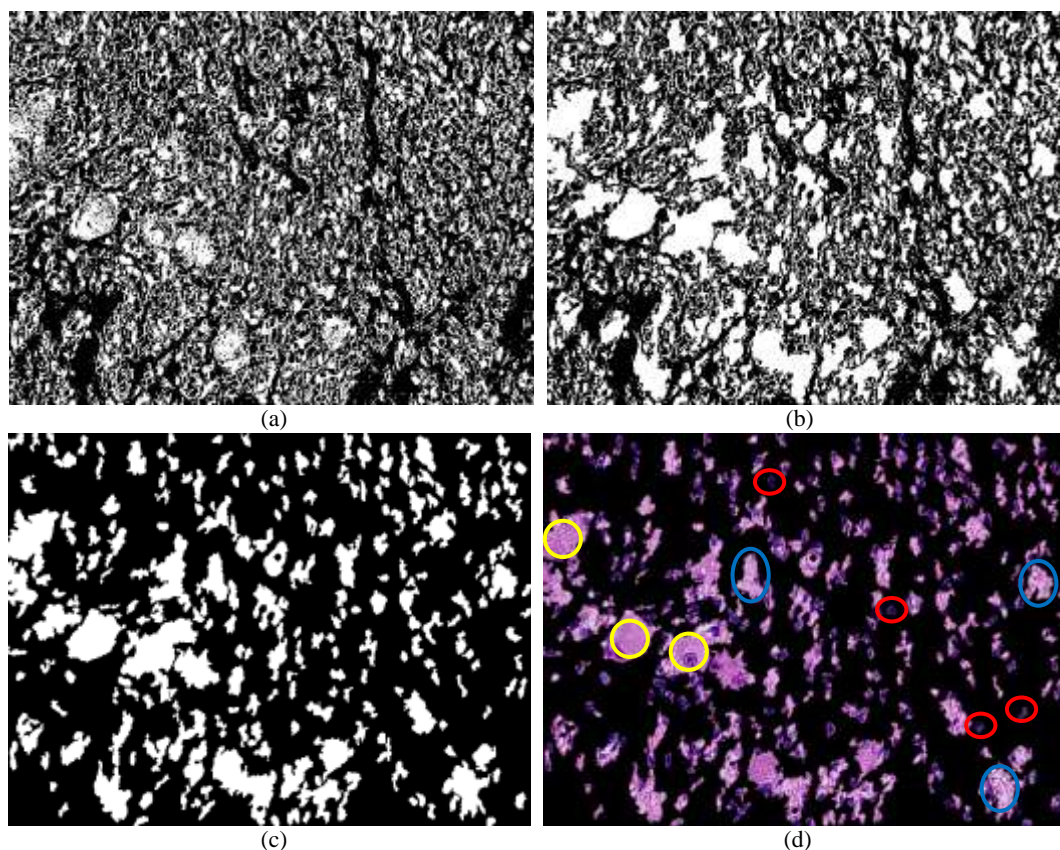


Figure 5.19 (a) After thresholding of saturation channel (>0.3) of the image in Figure 5.15, (b) After filling the region holes, (c) After eliminating blood cells and white areas, (d) After applying binary AND operation to the current binary image and the original color image.

One can observe in Figure 5.19 (d) that some of the differentiated cells (shown in yellow circles), some tumor cells (shown in red circles), and some undesired regions (shown in blue circles) are detected together. After the application of a thresholding with respect to areas of regions, the image in Figure 5.20 is obtained. When the image is examined, it is seen that the other tumor cells shown by red circles and undesired regions shown by green circles are detected. It is also seen that some white-blue regions exist within undesired regions indicated by green color. Although

the white-blue regions were eliminated from the image previously, they are again seen in the final image. This is because, after the holes are filled, the elimination of some white-blue regions is partially cancelled.

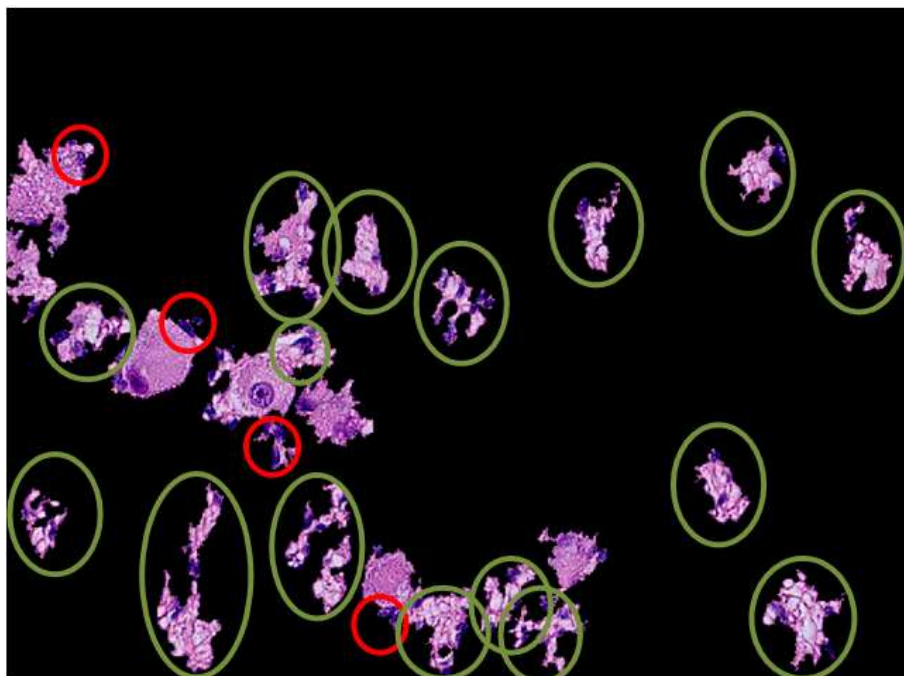


Figure 5.20 The other tumor cells (red circles) and undesired regions (green circles).

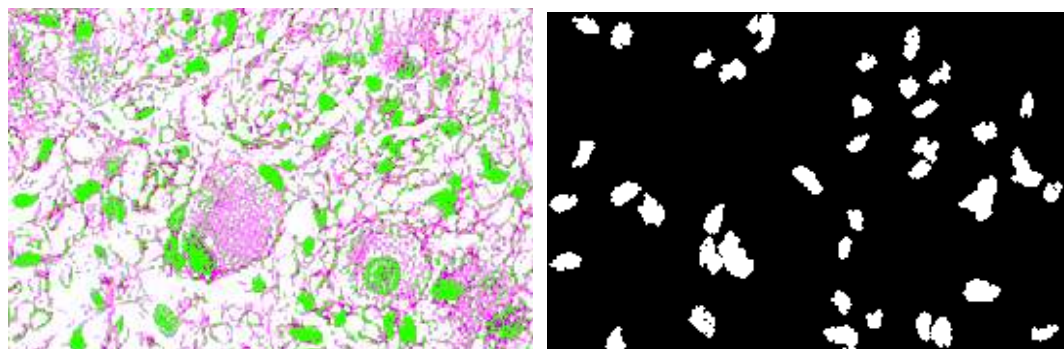


Figure 5.21 The detected dark purple cells (labeled with green) (left) the binary image for the other tumor cells (right).

The CODA algorithm is used for the detection of the other tumor (dark purple) cells in the contrast-enhanced color image (Figure 5.15). The minimum and maximum hue threshold levels are chosen as 0.5 and 0.8 respectively, and the saturation threshold is chosen as 0.5 for the CODA algorithm. Figure 5.21 shows the

detected dark purple cells (labeled with green) and the binary image which is the final output of the CODA algorithm.

The morphological operations and region features (eccentricity, area, and perimeter) are used to pre-process the image. The border cells are also eliminated as seen in Figure 5.22, and the resulting binary image for dark purple cells is obtained.

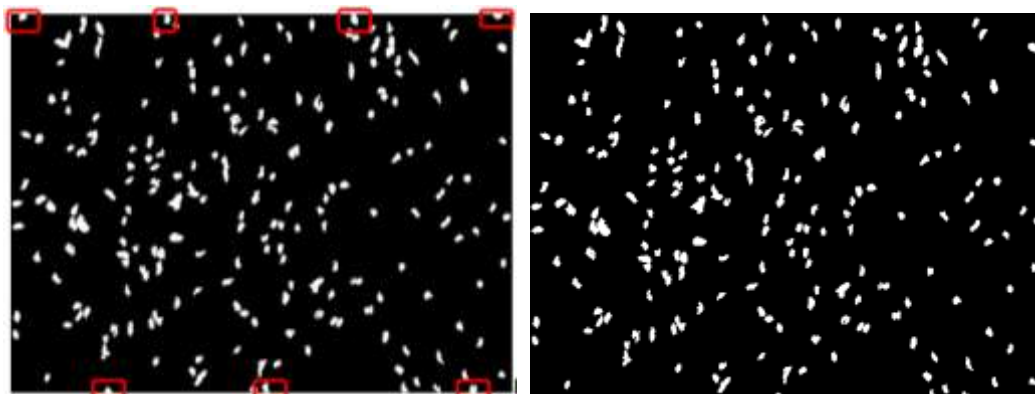


Figure 5.22 Elimination of border cells.

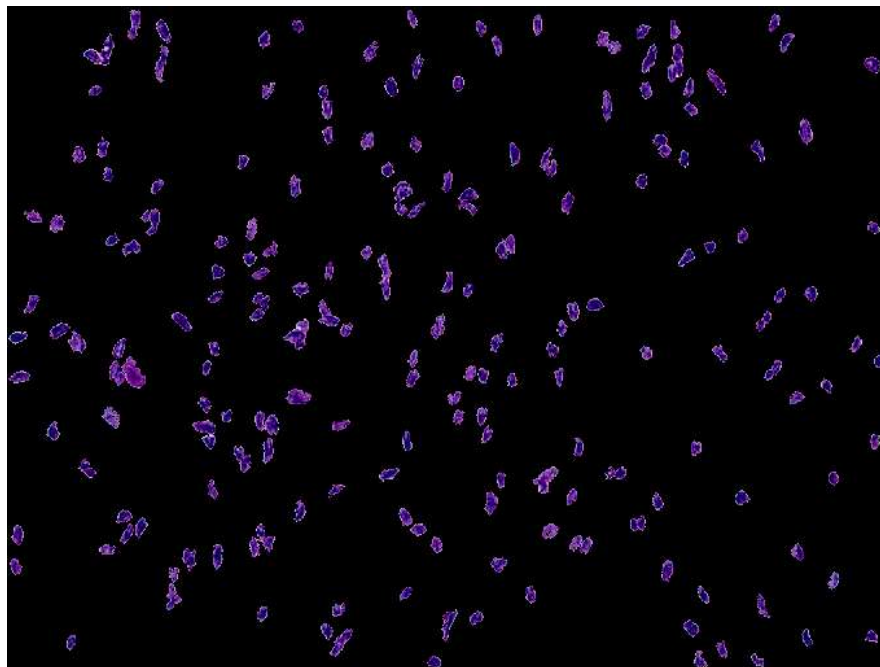


Figure 5.23 The image obtained after applying binary AND operation to the tumor binary image in Figure 5.22 and the original color image in Figure 5.14.

Now, the binary image given in Figure 5.22 is used for detecting tumor cells except the differentiated cells as seen in Figure 5.23.

Furthermore, although differentiated cells have larger cytoplasms in general, some differentiated cells might still have smaller cytoplasms. They are detected together with the other tumor cells, but they can be separated by using region characteristics. However, the image in Figure 5.23 does not include any such differentiated cell.

Binary AND operation is applied to the transpose of the binary image in Figure 5.22 and the image in Figure 5.20. Also, some morphological operations (erosion, opening) are used to eliminate some small regions. Figure 5.24 shows application of these steps.

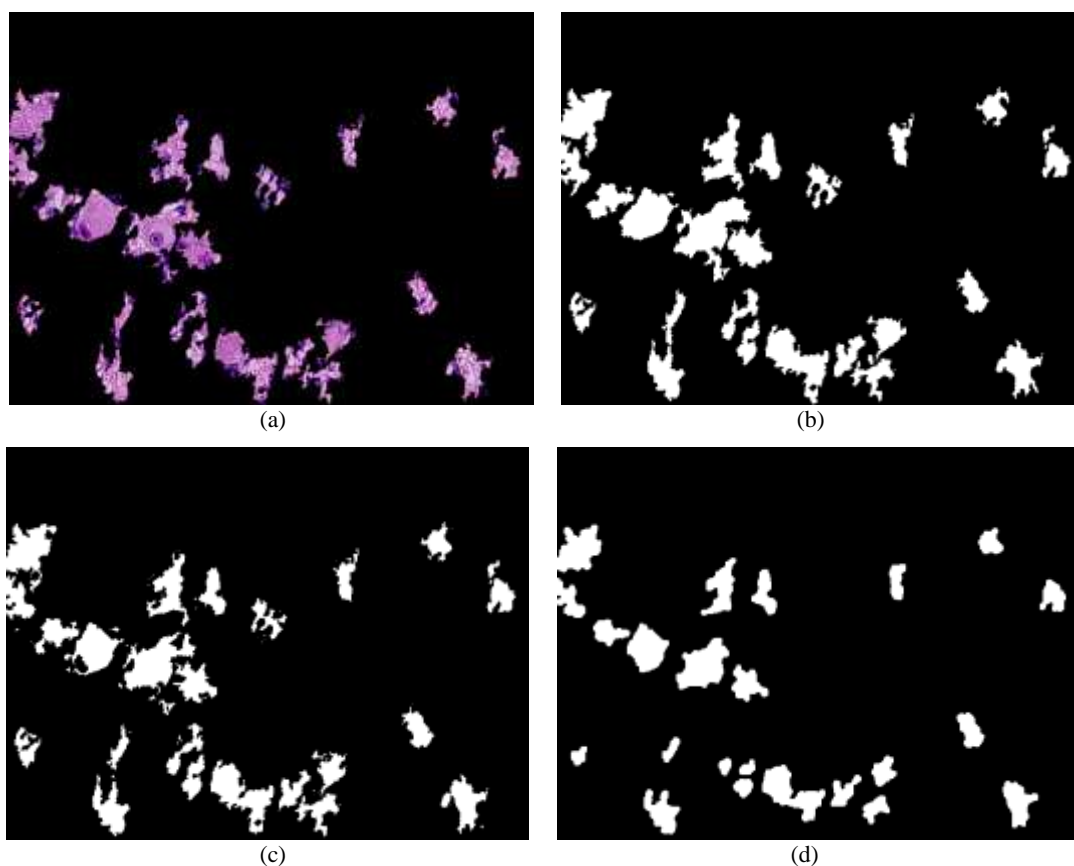


Figure 5.24 (a) The roughly detected differentiated cells together with the other tumor cells, (b) The binary image of (a), (c) The image after the elimination of the other tumor cells, (d) After the application of morphological operations.

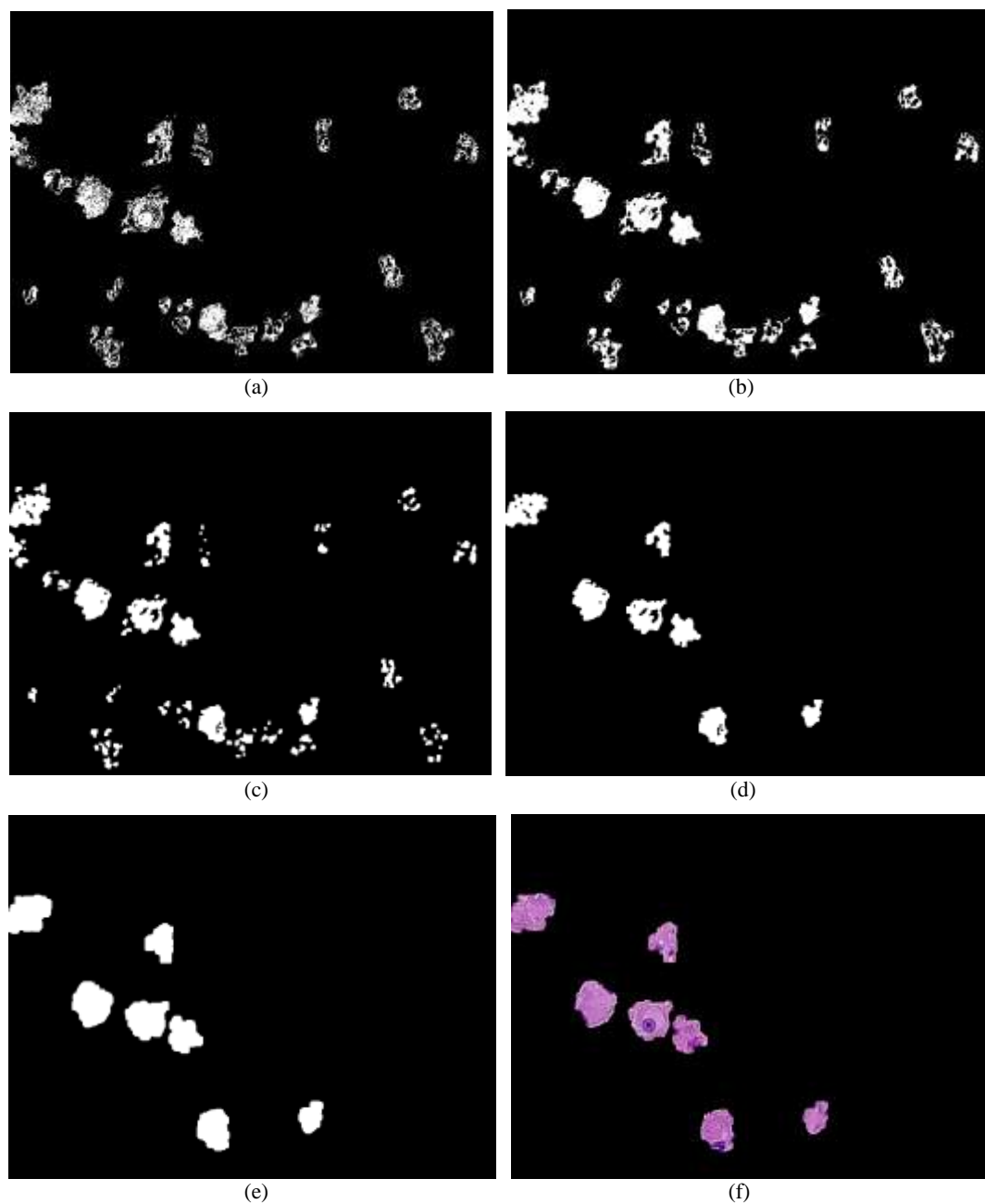


Figure 5.25 (a) The obtained image after applying binary AND operation to the transpose of white-blue binary image and Figure 5.24 (d), (b) After elimination of small regions and application of morphological operation (close) to (a), (c) After application of morphological operation (open) to (b), (d) After elimination of small regions, (e) After application of morphological operation (dilate) and filling the holes, (f) The obtained image after applying binary AND operation to (e) and the original color image.

Binary AND operation is again applied to the transpose of white-blue regions (Figure 5.17 (b)) and the image given in Figure 5.24 (d). In order to improve the binary mask image, some morphological operations are applied to eliminate the

undesired regions such as small regions and holes in the cells, as seen in Figure 5.25 (a), (b), (c), and (d), respectively. After all these operations, the binary masking image for differentiated cells are obtained as seen in Figure 5.25 (e) and also the masked color image is shown in Figure 5.25 (f).

Moreover, the images of the extracted differentiated cells and the other tumor cells are obtained separately as seen in Figure 5.26 (a) and (b), respectively, by using the CDDI algorithm. These images are to be further used in the feature extraction and classification stages.

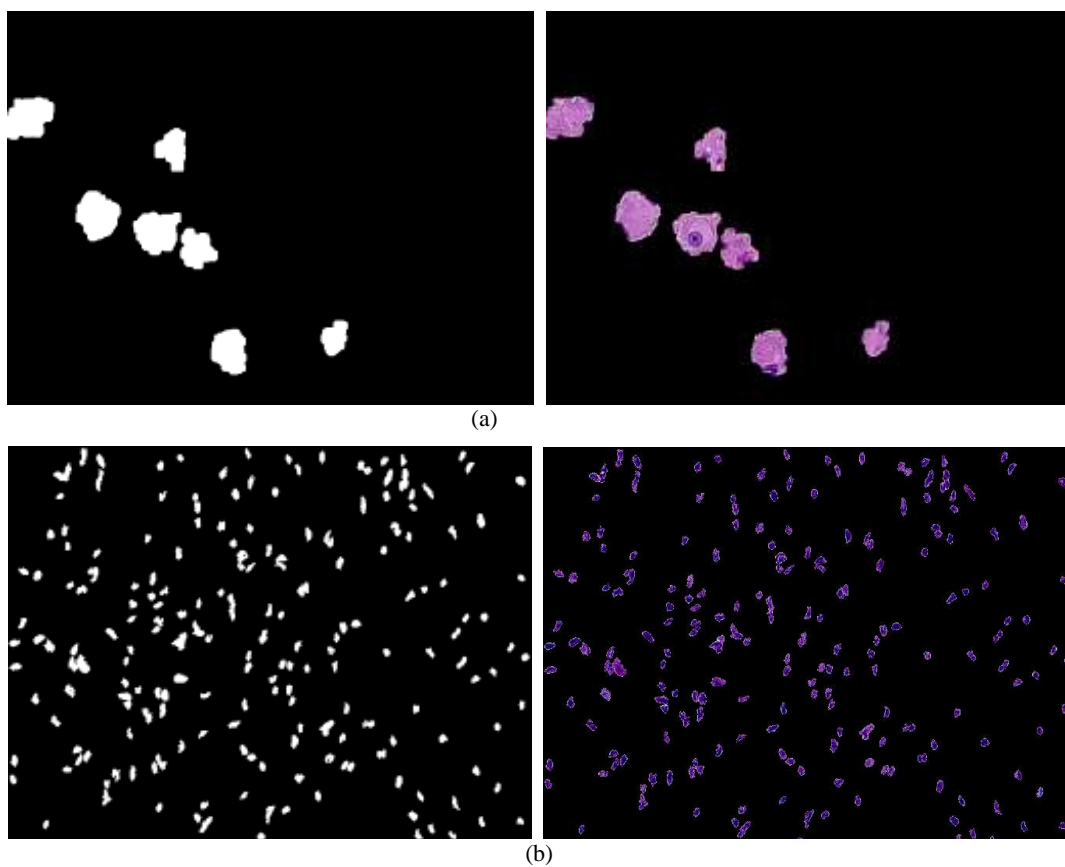


Figure 5.26 (a) The binary mask for the differentiated cells (left) and the image after applying binary AND operation to the binary mask and the original color image (right), (b) The binary mask for the other tumor cells (left) and the image after applying binary AND operation to binary mask and the original color image (right).

5.3 Detection of Cells in the Images for Determining Mitosis Karyorrhexis (MK) Index

In this section, the structure of the CDUI algorithm is explained in detail. If neuropil percentage of 20x zoomed image is lower than 5%, the tissue is determined as of undifferentiated type, so that new image which is taken under 100x magnification must be used as the input of the CDUI algorithm to create a binary image for mitosis and karyorrhexis cells and the other tumor cells. The flowchart of the CDUI algorithm is shown in Figure 5.27.

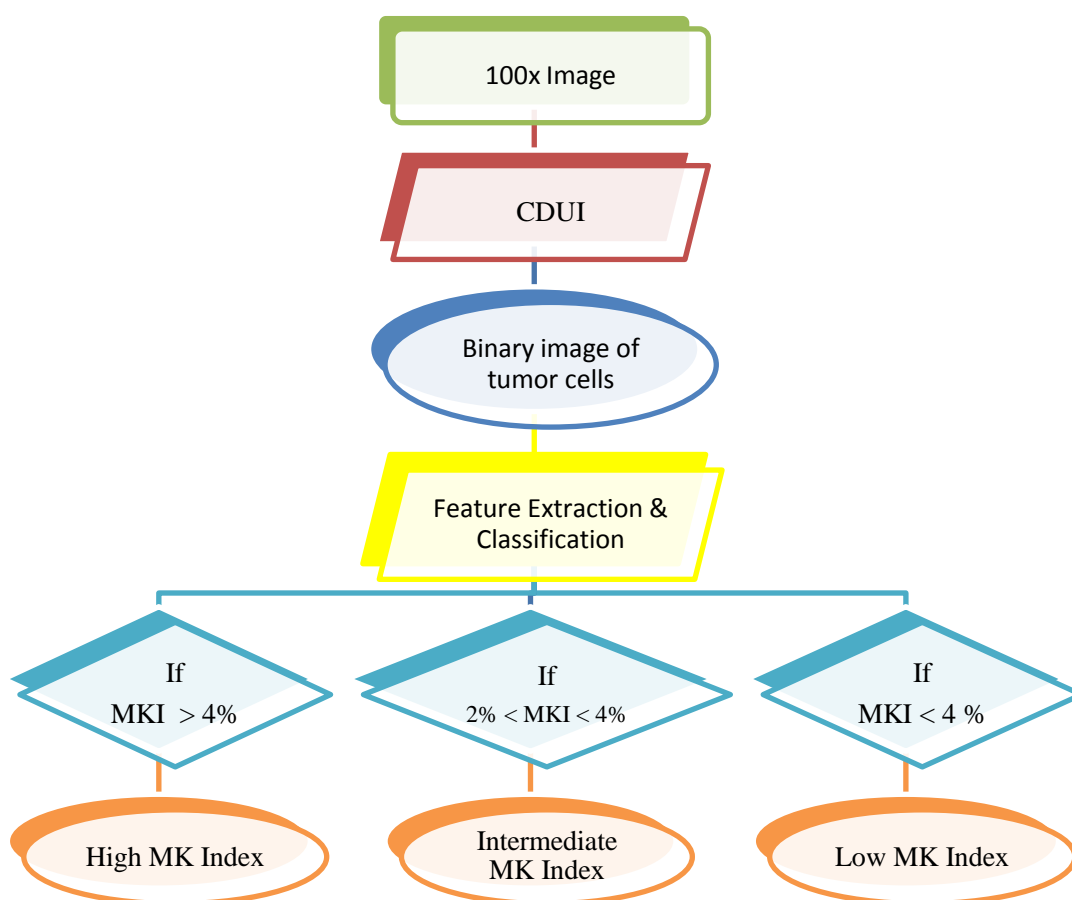


Figure 5.27 The flowchart of the CDUI algorithm.

Figures 5.28 and 5.29 show two images of undifferentiated type belonging to two different patients. Hue, saturation, and value channels of the images can also be seen in both figures. In the following, the output of the CDUI algorithm will be determined and shown for both patients.

The aim of the CDUI algorithm is to detect the tumor cells, same as the CDDI algorithm. In order to detect the cells, the color thresholding method is used again as in the CDDI algorithm. If Figure 5.28 (a) and Figure 5.29 (a) are compared, it is realized that the tissue color distributions and color scales are different. Because of this, the saturation threshold level is determined by averaging the pixel values of the saturation channel. By this way, a dynamic threshold level can be calculated based on the image.

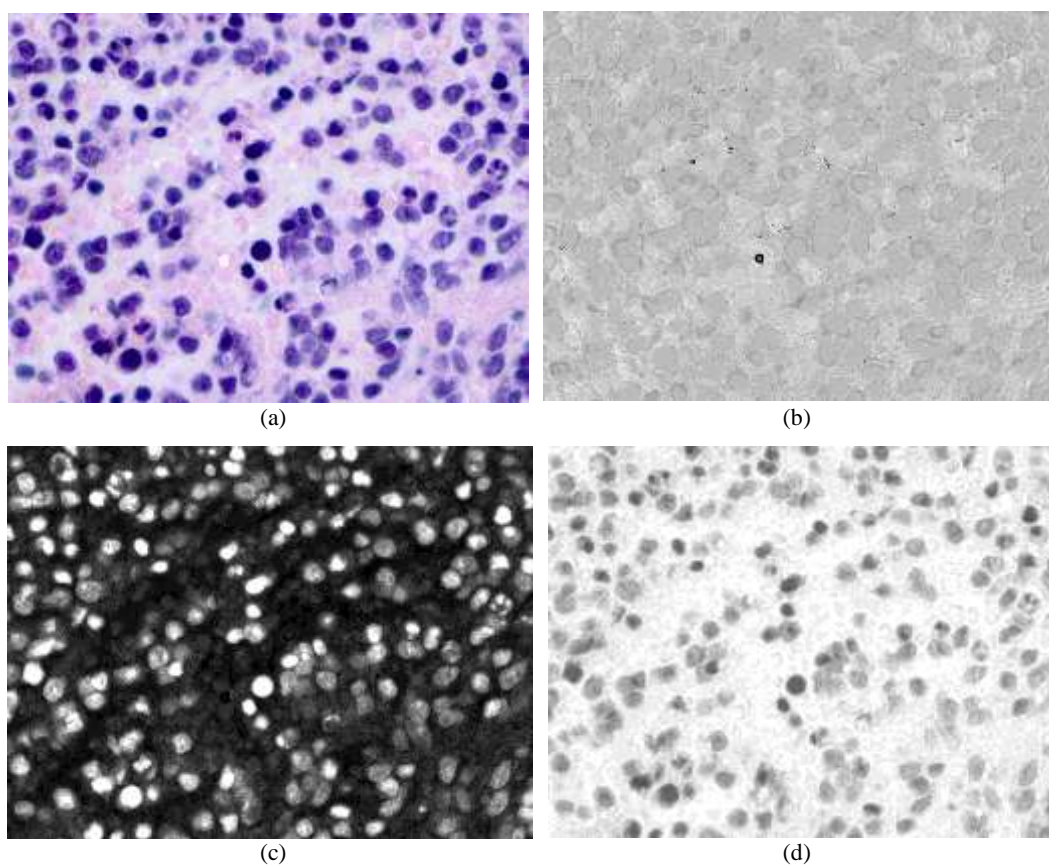


Figure 5.28 (a) Tissue image of NB tumor with 100x zoom for Patient 1. (b) Hue channel of the image, (c) Saturation channel of the image, (d) Value channel of the image.

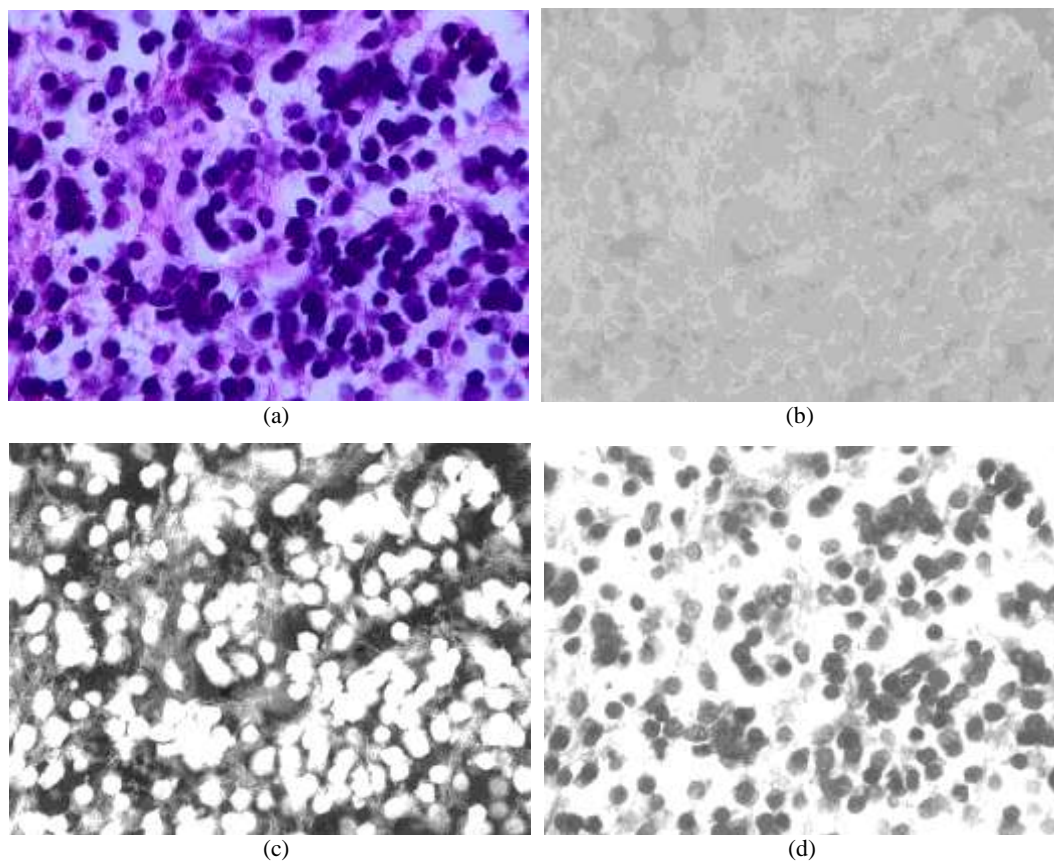


Figure 5.29 (a) Tissue image of NB tumor with 100x zoom for Patient 2. (b) Hue channel of the image, (c) Saturation channel of the image, (d) Value channel of the image.

After the values of the pixels in the value channel are set to one, the color images given in Figure 5.30 is obtained for two different patients.

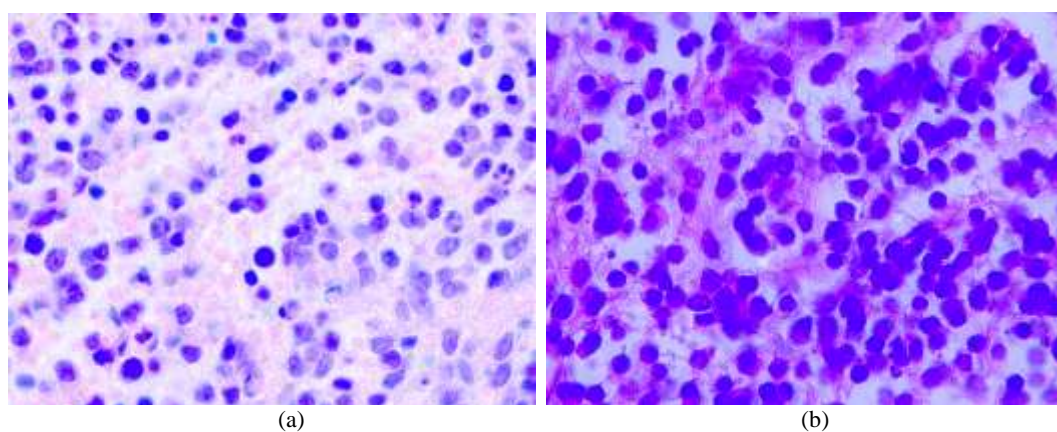


Figure 5.30 The color images when value channels of the images are set to one; (a) Patient 1, (b) Patient 2.

Again, by using the color information, the binary mask image to be used for masking the tumor cells is created. For this purpose, the CODA algorithm can be used. The images have different saturation cutoff levels; so that the saturation threshold level is calculated automatically and the binary mask image is obtained. Most of the tumor cells are detected as seen in Figure 5.31 where the labeling color is orange.

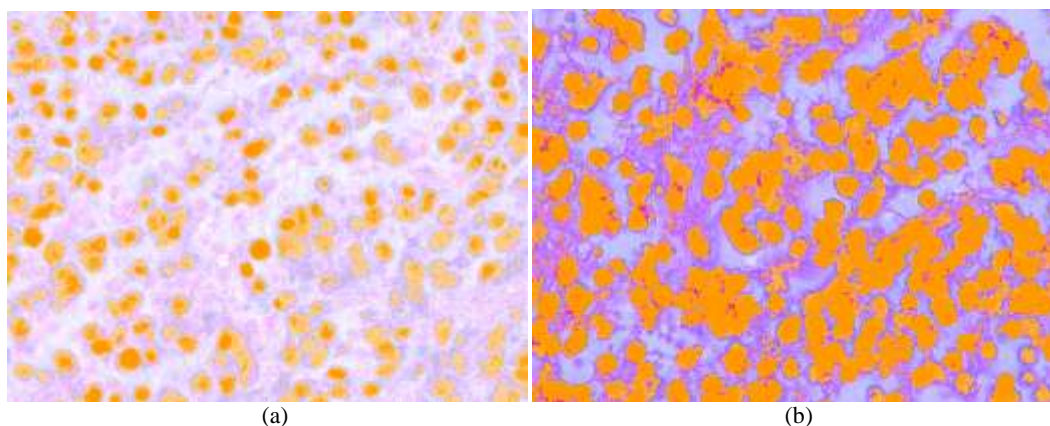


Figure 5.31 Detected tumor cells (labeled with orange); (a) Patient 1, (b) Patient 2.

The rough binary mask is obtained now. As seen in Figure 5.32, there are undesired small regions. In order to improve the binary mask image some morphological operations (opening, closing) are applied to eliminate these undesired regions such as small regions and holes in the cells.

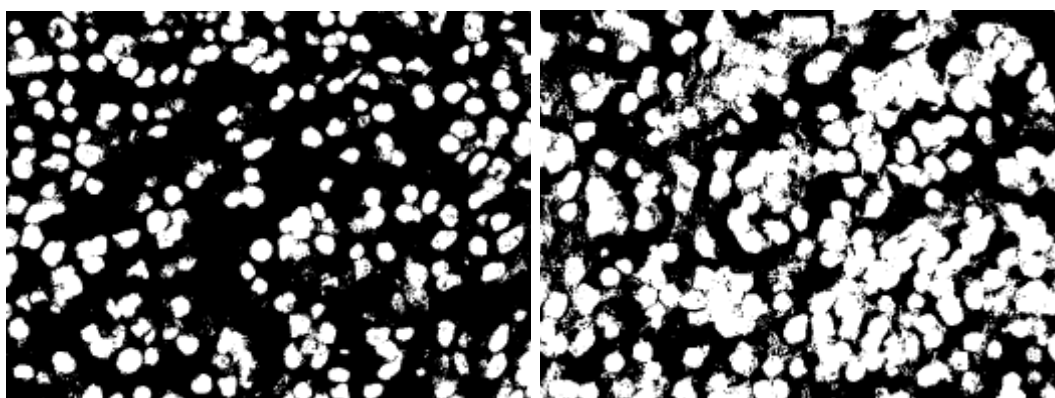


Figure 5.32 The initially obtained binary mask images for both patients.

In order to improve the rough binary image, closing operation is used. The newly obtained images are shown in Figure 5.33.

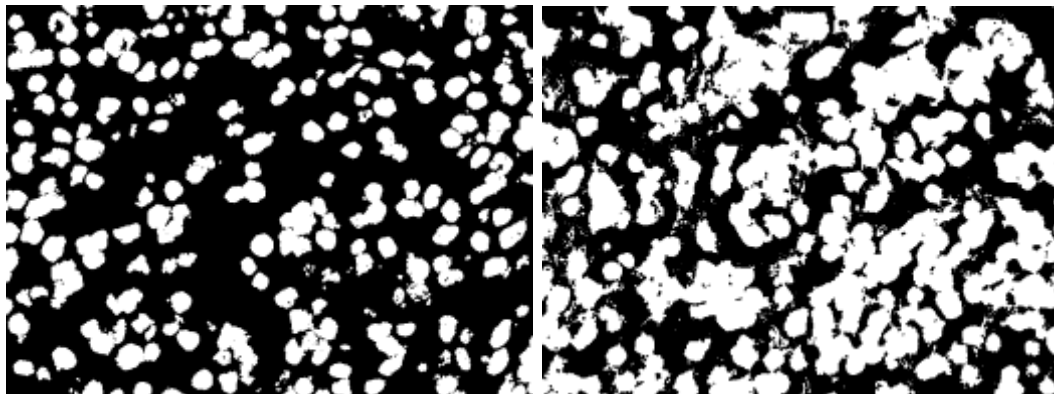


Figure 5.33 Binary images after the application of closing operation.

The images are still not useful, because there are many small areas and unwanted holes in the detected areas. For this reason opening operation is applied. Thus, holes are filled and also small areas are eliminated as seen in Figure 5.34.

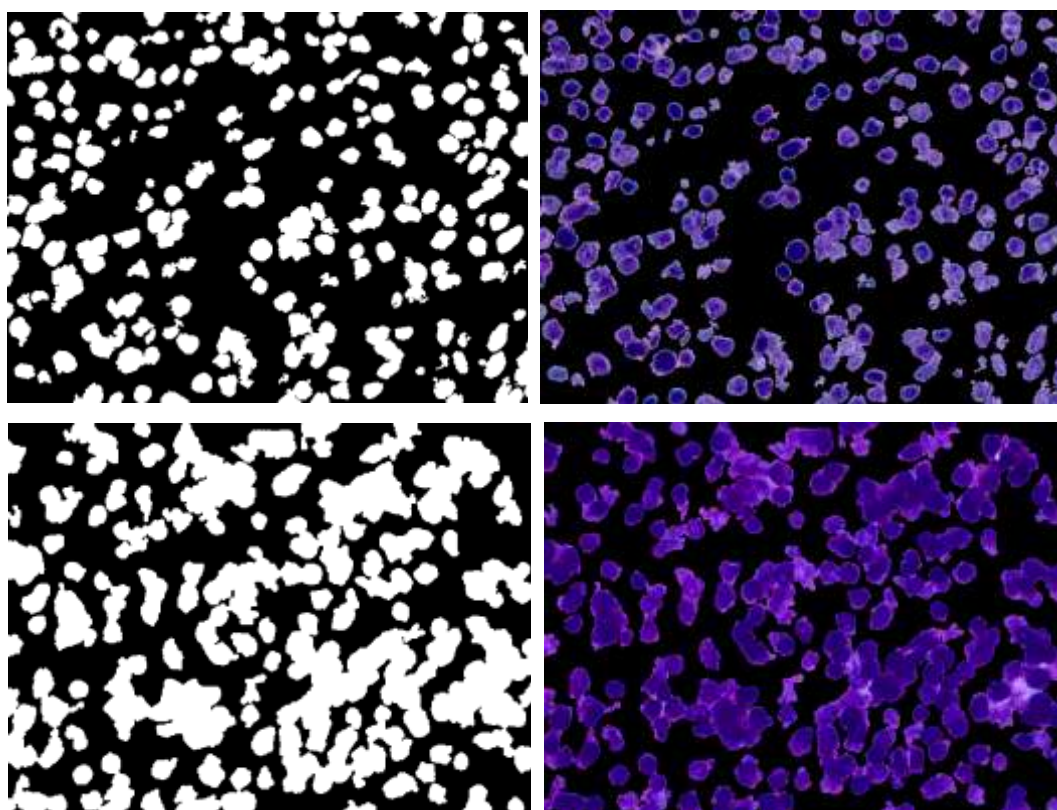


Figure 5.34 The binary images after elimination of small areas and holes and after applying binary AND operation to the masks and original images.

As shown in Figure 5.35, most of the cells are detected as attached to each other. Hence, it is not appropriate to extract features and perform classification by using these binary images. Because of this, a type of watershed algorithm described in Meyer, F. (1994) is used in order to separate these attached cells.

The first step of our watershed algorithm is the labeling of independent regions in the image. The area values of the labeled regions are examined and the following operations are performed for the regions that have areas larger than a predetermined threshold value.

- Compute the distance transform (Jain, A., 1989) of the binary image complement.
- Complement the distance transform, and force pixels that do not belong to the objects to be at $-Inf$.

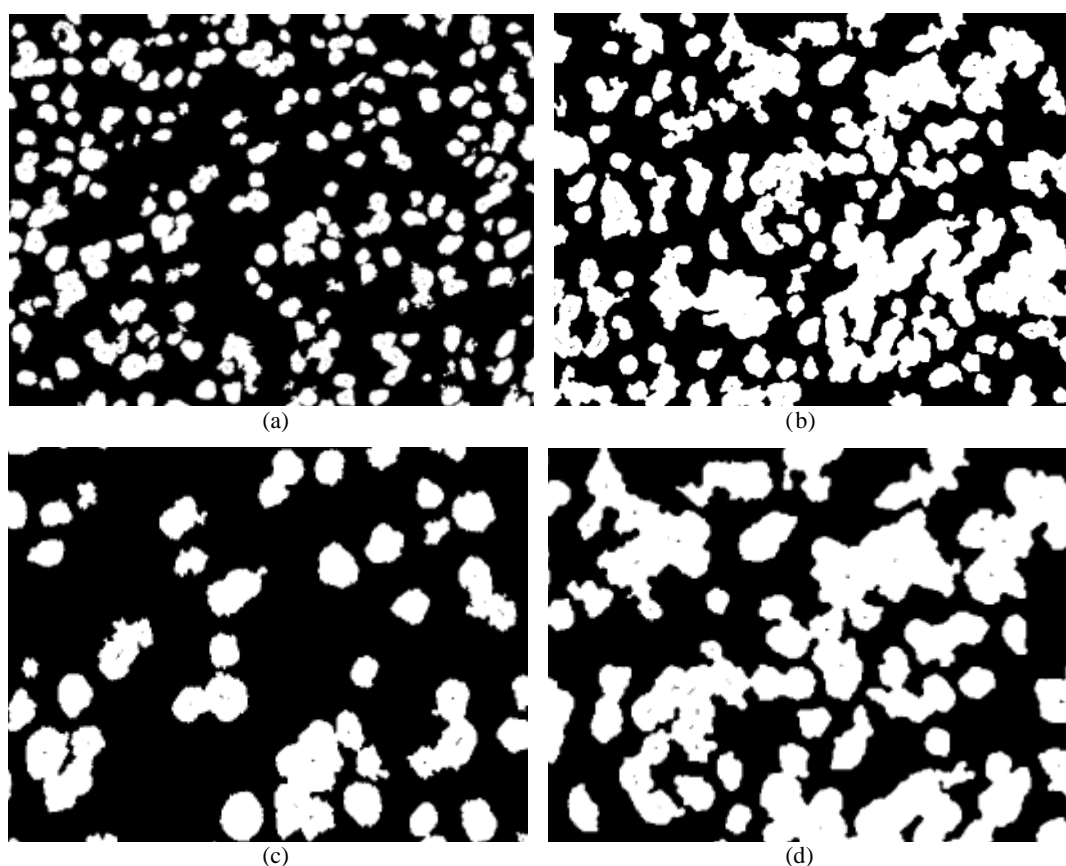


Figure 5.35 (a), (b) The binary images including regional minimum points, (c), (d) After zooming in the images given in (a) and (b), respectively.

Then, find the regional minimum points in the image and process them as shown in Figure 5.35.

Erosion operation is applied by using the minimum points given in Figure 5.35. By this way, if there are several points for a single cell, they are combined to create the cell (Figure 5.36 (a), (b)).

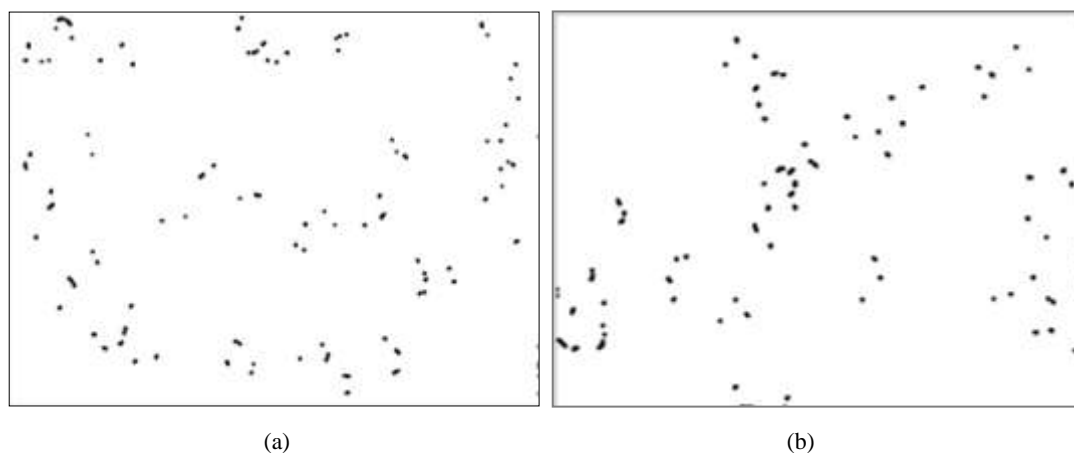


Figure 5.36 (a), (b) The minimum points obtained after eroding process.

The midpoints of each black region are determined and the squares having a pre-determined size are placed on these midpoints (Figure 5.37 (a) and (b)). However, this operation is not applied to regions having areas under a certain value, because the possibility to find several cells in such small regions is too low. Then, the inverse of the image is calculated by converting the original colored image into gray-level image (Figure 5.37 (c), (d)). Finally, these images are combined with the binary images including squares by using *imimposemin* command and the images in Figure 5.37 (e) and (f) are obtained. These final images are used as input of the watershed algorithm.

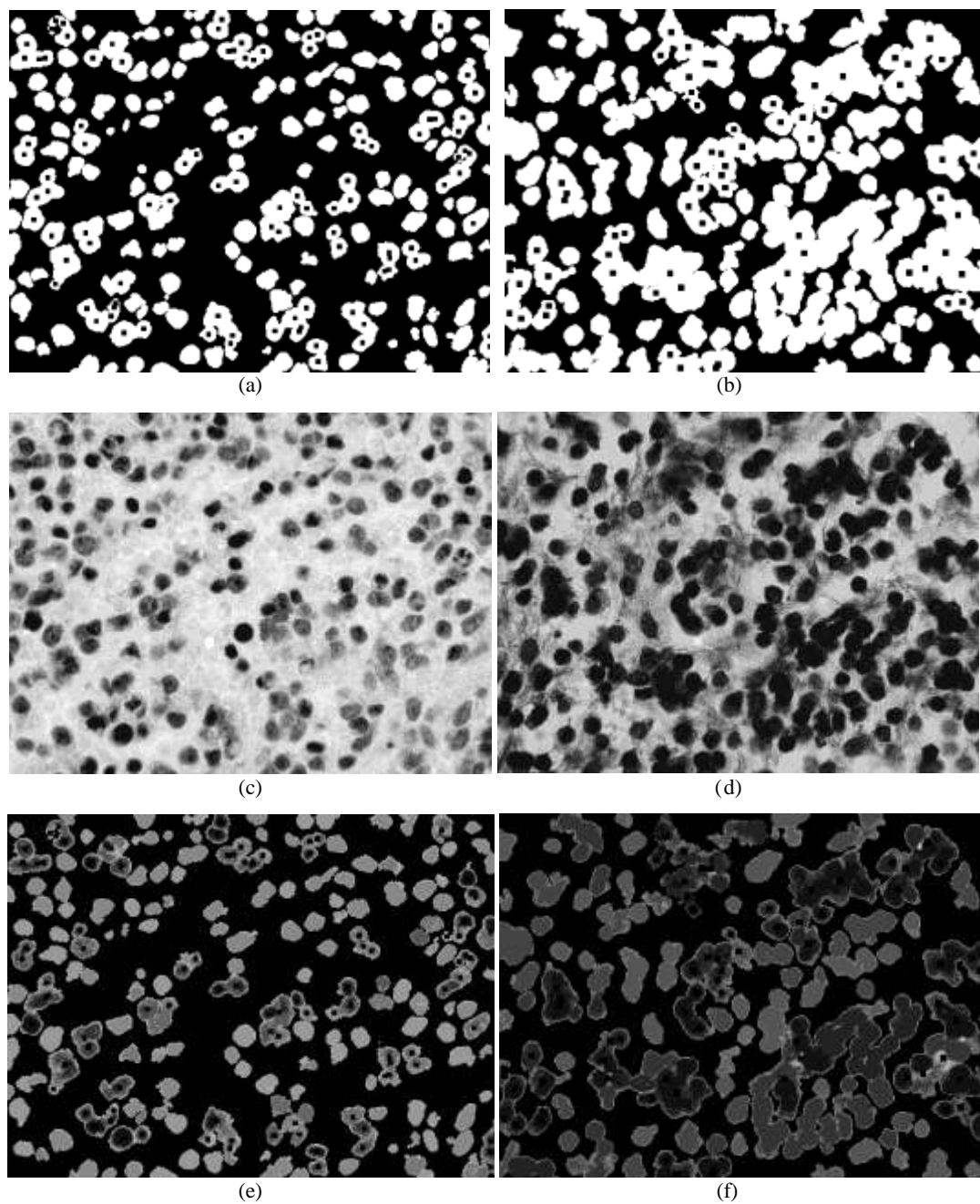


Figure 5.37 (a), (b) The binary images with inserted squares, (c), (d) The inverse of the gray-level image of the original image, (e), (f) The combined images by applying *imimposemin* command to the images (a) and (c) with (b) and (d), respectively.

The binary images obtained after the application of the watershed algorithm are shown in Figure 5.38. If the images are examined, it is seen that there are regions including more than one cell. It is observed that the detection of MK cells in the image on the left hand side of Figure 5.38 is better although all the cells are still not separated after the application of the watershed algorithm.

After the removal of components on image borders and elimination of small and large regions, the new binary images are obtained as shown in Figures 5.39 and 5.40.

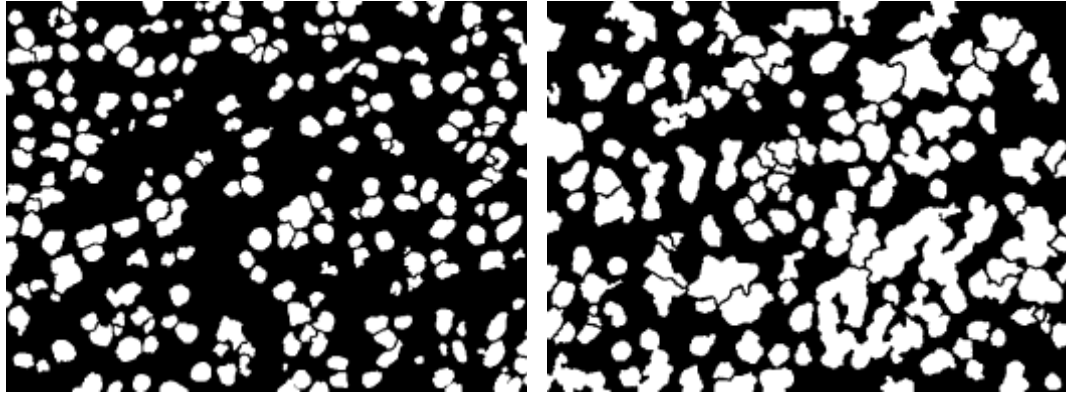


Figure 5.38 The binary images after the application of the watershed algorithm.

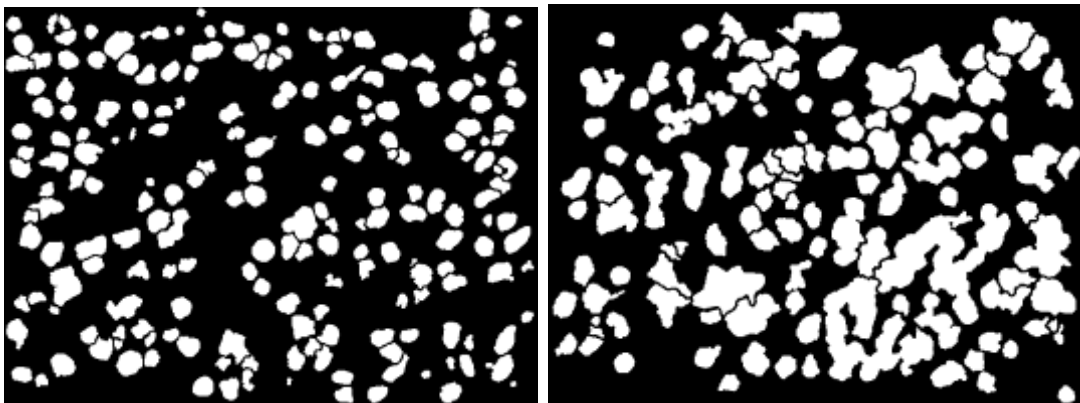


Figure 5.39 The binary images after elimination of components on image borders.

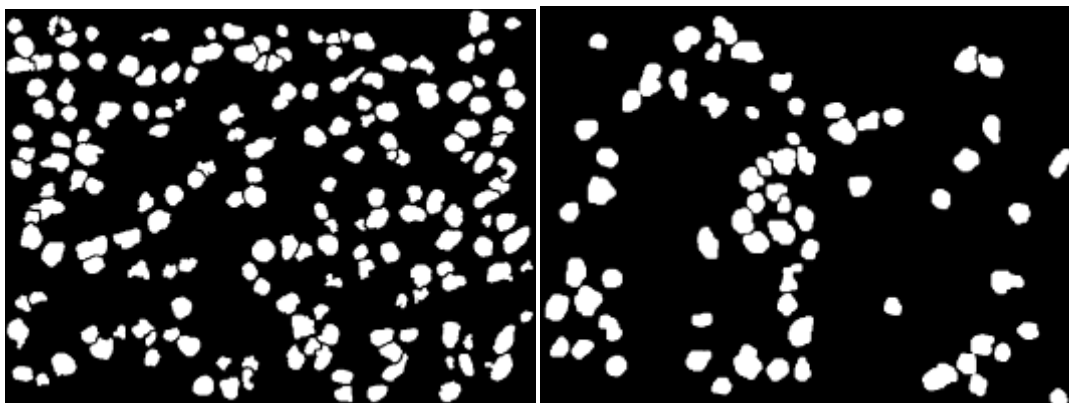


Figure 5.40 The binary images after elimination of small and large regions.

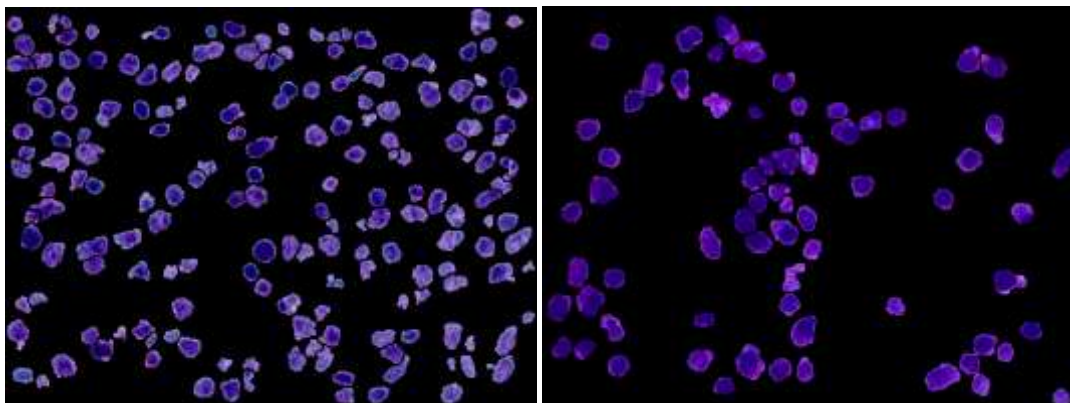


Figure 5.41 The final images obtained after applying binary AND operation to the binary images in Figure 5.40 and the original images in Figures 5.28 (a) and 5.29 (a).

The binary images to be used for feature extraction are obtained as shown in Figure 5.40. Binary AND operation is applied to the original colored images and the binary images in Figure 5.40 and the final images shown in Figure 5.41 are obtained.

In the next section, separate images are obtained for each cell and then region and texture features are calculated for each image. Next, the classification is performed by using the artificial neural network structure.

5.4 Feature Extraction

Up to now, operations are applied to whole images. In order to extract the features belonging to the cells, the cells must be segmented from the image. In order to extract the cells from the whole image, “Binary Cell Image (BCI)” algorithm is written. The block diagram of that algorithm is given in Figure 5.42.

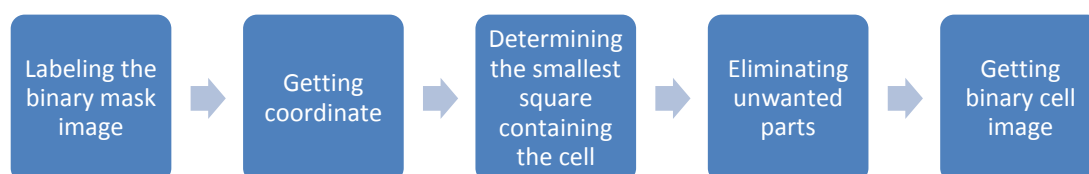


Figure 5.42 The block diagram of the BCI algorithm.

If the binary image is labeled, the coordinates of the cells are found separately. In MATLAB, an image can only be defined as having a quadrilateral shape, so that the

smallest quadrilaterals are found for all the detected cells by using the coordinate data. After that, the cells are obtained as small images. However, other cells close to the labeled cell can create problems as seen in Figure 5.43 (red rectangular). These unwanted parts must be eliminated and the image must only include the selected cell. After eliminating the unwanted parts, the binary cell images are obtained as given in Figure 5.44. A MATLAB function called *Binary Cell Image* forms this part of the algorithm. The input of the algorithm is the binary image and the outputs of the algorithm are the binary cell images. The purpose of this function is to extract cells from the whole image.

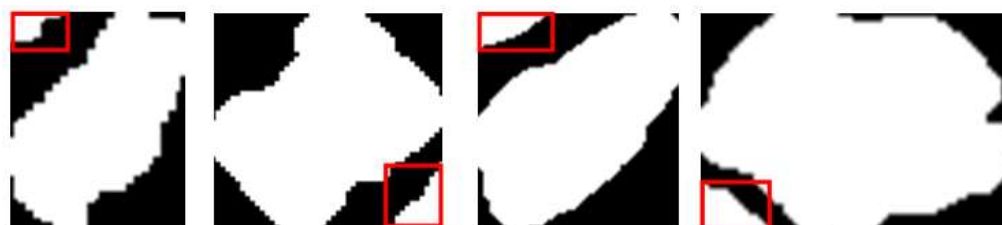


Figure 5.43 The binary cell images with unwanted parts.



Figure 5.44 The binary cell images after the elimination of unwanted parts.

The correct segmentation of each cell is crucial in order to obtain a useful feature matrix for the classification of the cells. Our aim is not only the segmentation of the cell regions, but also to count and classify them as mentioned before.

In order to obtain region features of the cells, the binary image must be labeled with *bwlabel* command. After that, by using *regionprops* command, area, perimeter, eccentricity, and convexarea are calculated for all the extracted cells.

For texture analysis of the cells, features obtained by Haralick's co-occurrence matrix are used. The input of the Haralick function must be a grayscale image, so

that the masked color image is converted into grayscale image by using *rgb2gray* command (Figure 5.45). Then, the cells are obtained from the whole image by using BCI algorithm (Figure 5.46 (a), (b)). The texture features depend on the pixel values in the image. As seen in Figure 5.45, the background of the cell image is black and the cells do not have the same dimensions. The background can affect the values of the obtained features with each cell having different amount of background. Therefore, the pixels having zero values (black color) have been converted into *NaN* before the calculation of the co-occurrence matrix.

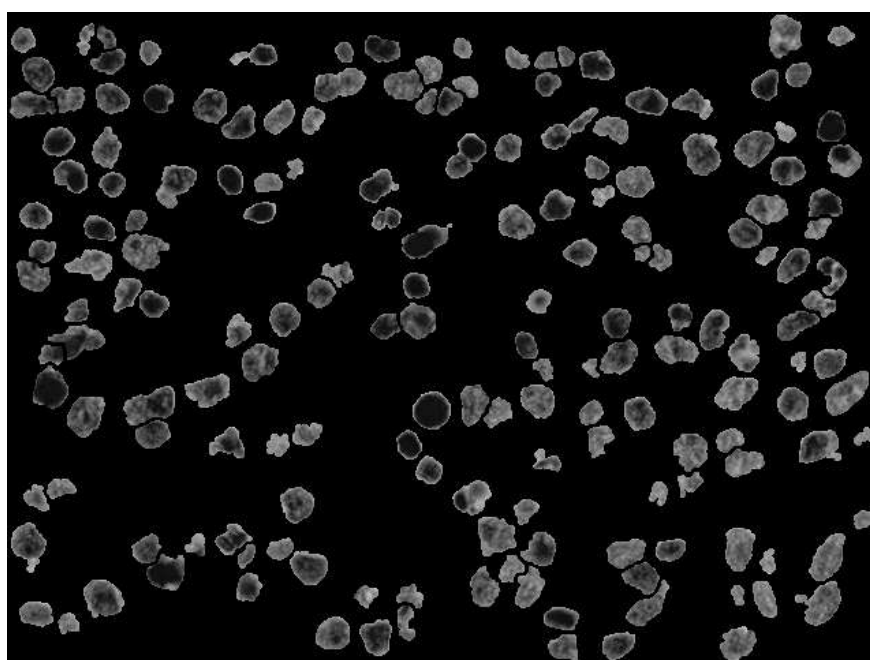


Figure 5.45 The masked color image after conversion of the image into grayscale.



(a)



(b)

Figure 5.46 Creating input data for the Haralick function (a) The binary cell mask image (b) The grayscale cell image.

A gray level co-occurrence matrix (GLCM) is created by using *graycomatrix* using grayscale images as in Figure 5.46 (b). *GLCM_Features1* helps to calculate features from different GLCMs that are input to the function. After running *GLCM_Features1* function, we obtain a 20xN matrix where 20 is the number of Haralick features to be used and N is the number of cells in the image.

Perimeter, eccentricity, and convex area features are obtained from the binary image by using *regionprops* command. Besides, we calculate one more feature; it is the area of a cell, which does not contain any darkest purple region, divided by the cell area. The feature is named as *Non-darkest area*. As a result, the number of total features is 24.

Non-darkest area feature is found by using CODA, but the threshold levels are changed in order to accommodate different color ranges. At this step, the new binary mask image is created (Figure 5.47).

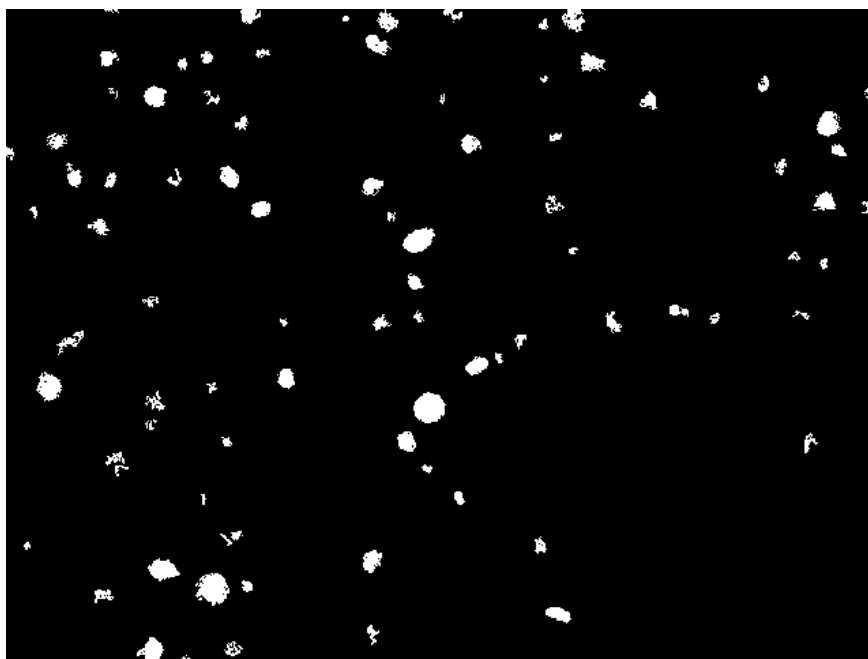


Figure 5.47 The binary mask image for darkest purple regions.

After applying binary AND operation to the binary mask image (Figure 5.40 (left)) and the transpose of the new binary mask image in Figure 5.47, some regions

disappear, because some tumor cells have only darkest purple color (Figure 5.48). All the area features of the cells are recalculated and area itself is used as a feature.

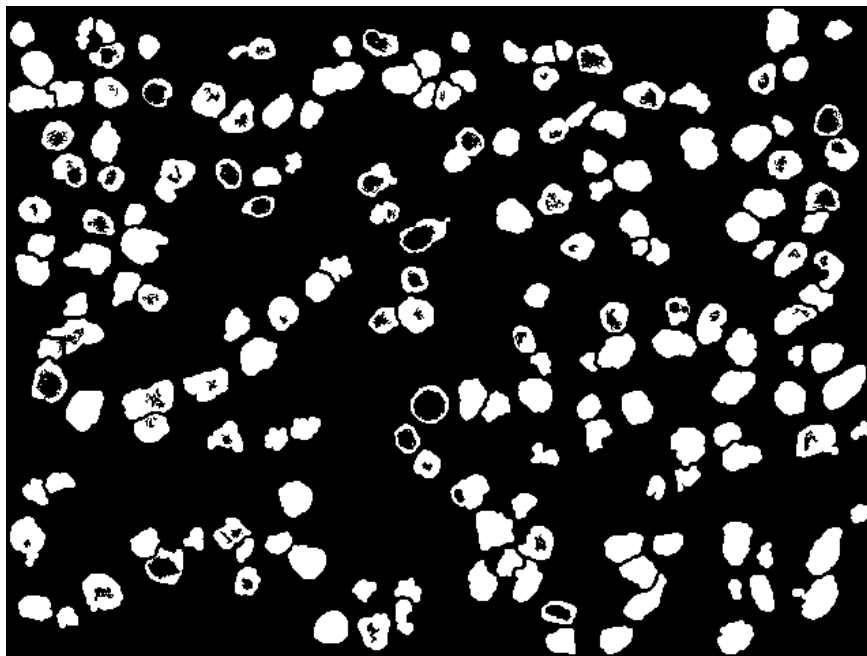


Figure 5.48 The image obtained after applying binary AND operation to the binary mask image in Figure 5.40 (left) and the new binary mask image in Figure 5.47.

The texture and region features of tumor cells are listed in Table 5.1 below.

Table 5.1 Texture and region features of tumor cells.

1. Autocorrelation	13. Sum Variance
2. Contrast	14. Sum Entropy
3. Correlation	15. Difference Variance
4. Cluster Prominence	16. Difference Entropy
5. Cluster Shade	17. Information Measurement of Correlation 1
6. Dissimilarity	18. Information Measurement of Correlation 2
7. Energy	19. Inverse Difference Normalized (INN)
8. Entropy	20. Inverse Difference Moment Normalized
9. Homogeneity	21. Perimeter
10. Maximum Probability	22. Convex Area
11. Sum of Squares: Variance	23. Eccentricity
12. Sum Area	24. Non- darkest Area

5.5 Classification

In order to classify the cells, we use artificial neural networks (ANNs) and ensemble methods in MATLAB.

5.5.1 *Input Data for Classification*

The feature matrices to be used for classification are obtained from the images in Figures 5.49 through 5.51.

In Figure 5.49, the images obtained from the tissue sample including MK index are shown. The mitosis and karyorrhexis cells are indicated by red circles on the original images given on the left hand side. The mitosis and karyorrhexis cells are again shown by red circles and the eliminated mitosis and karyorrhexis cells during the image processing operations are indicated by yellow circles on the processed images given on the right hand side. The feature matrix is created by calculating the features related to each cell given on the processed images on the right hand side. The obtained feature matrices are used for training ANNs and ensemble methods. In Figure 5.50, the images obtained from the tissue samples taken from another patient for MK index analysis can be seen. There are no mitosis and karyorrhexis cells in this set of images.

Images to be analyzed for grade of differentiation subtype are shown in Figure 5.51. The red circles on these images indicate the differentiated cells.

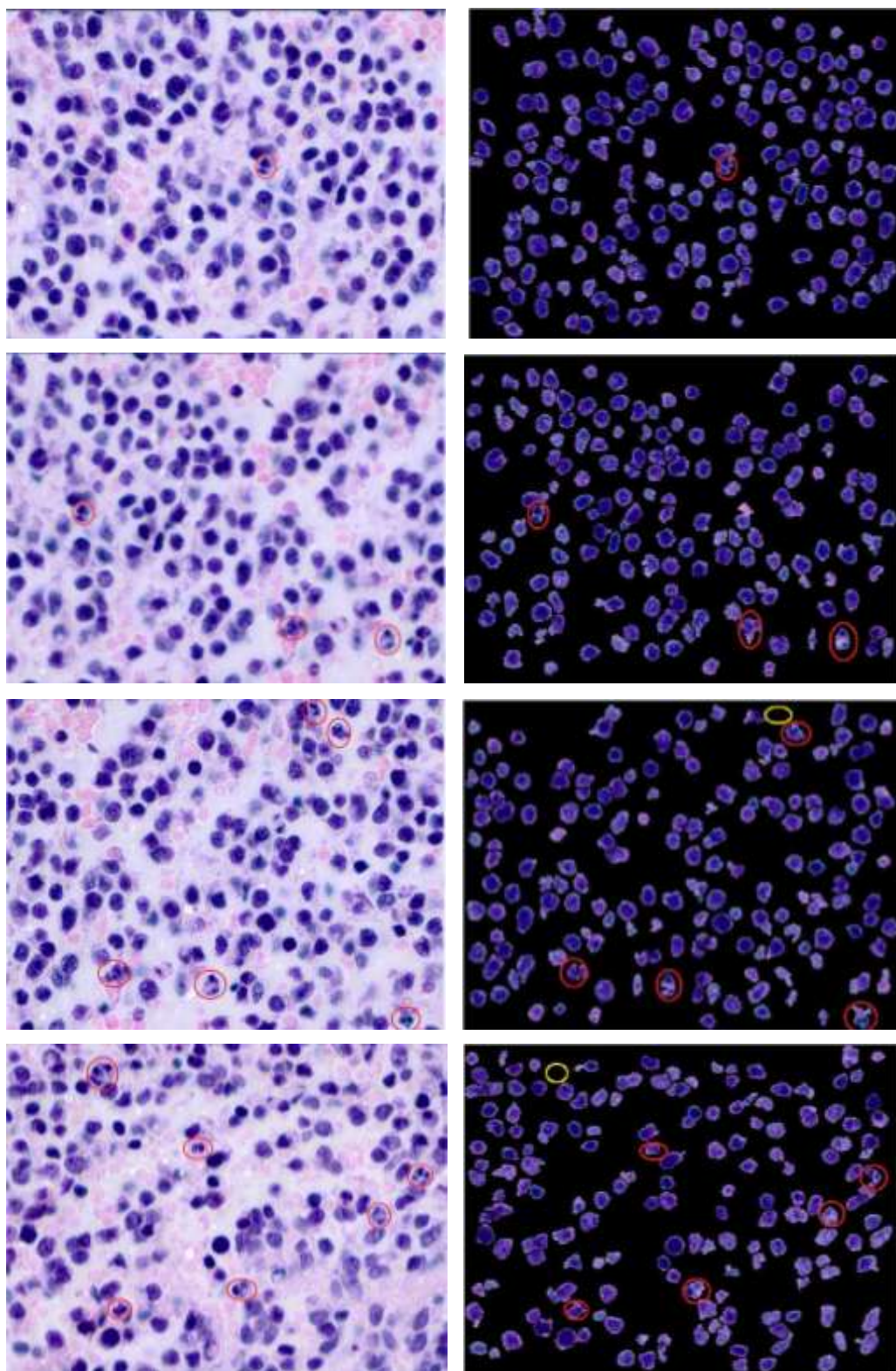


Figure 5.49 The images to be used in training set. The mitosis and karyorrhexis cells are indicated with red circles (left), the images obtained after the application of previously described operations to the original images (right).

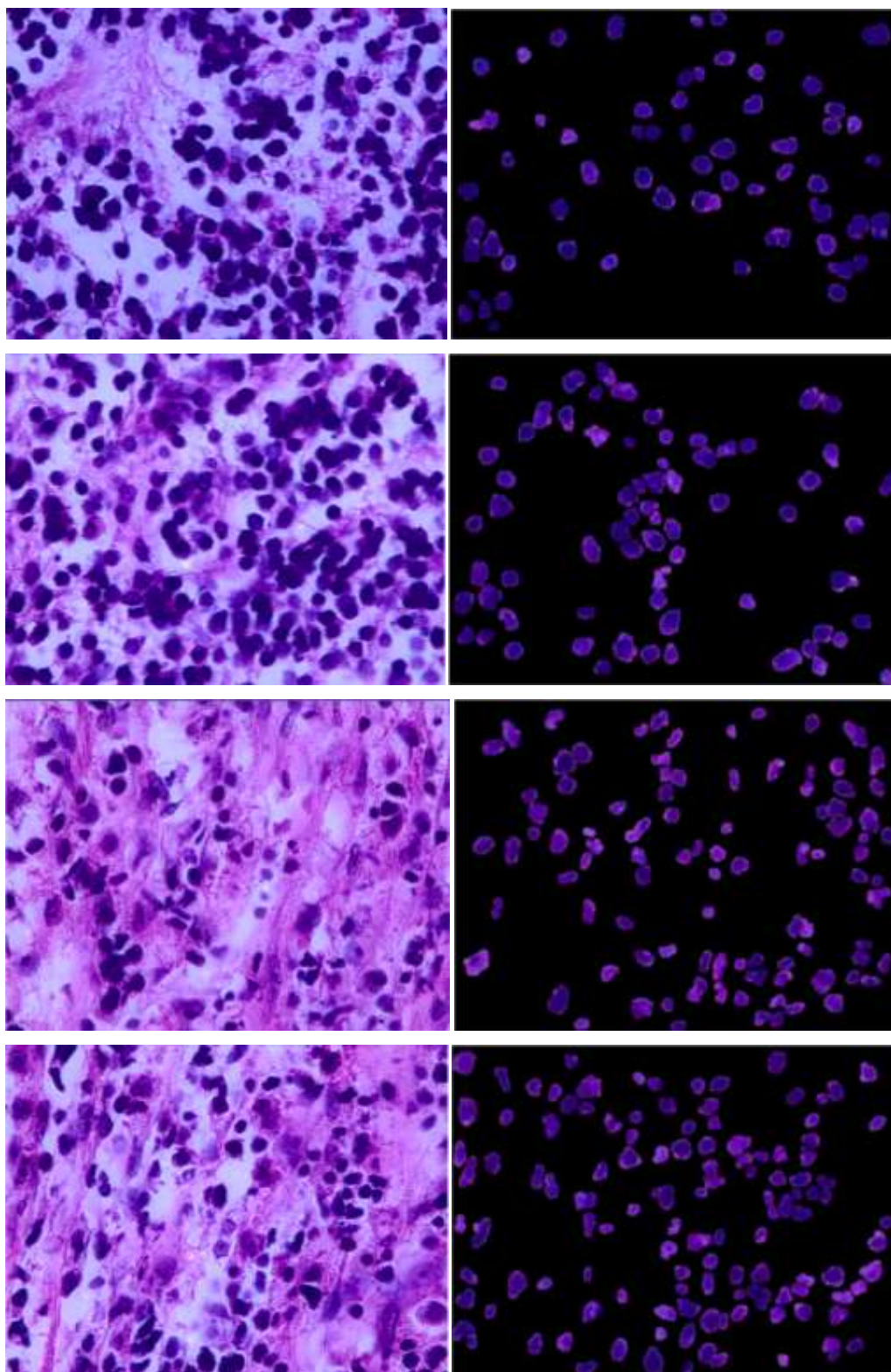


Figure 5.50 The original images to be used for training set (left), the images obtained after the application of previously described operations to the original images (right).

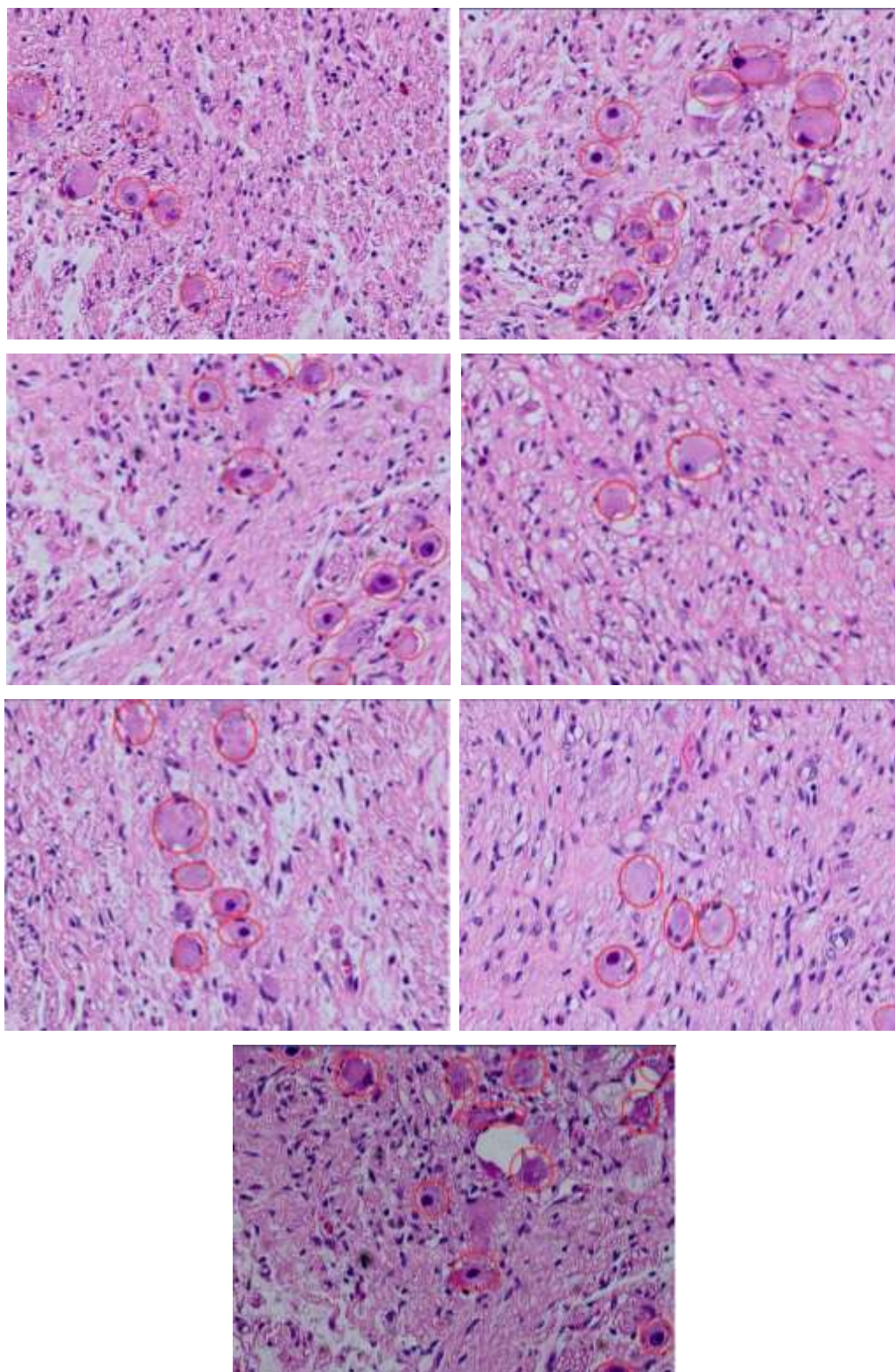


Figure 5.51 The images to be used as part of the training set. The differentiated cells are indicated with red circles.

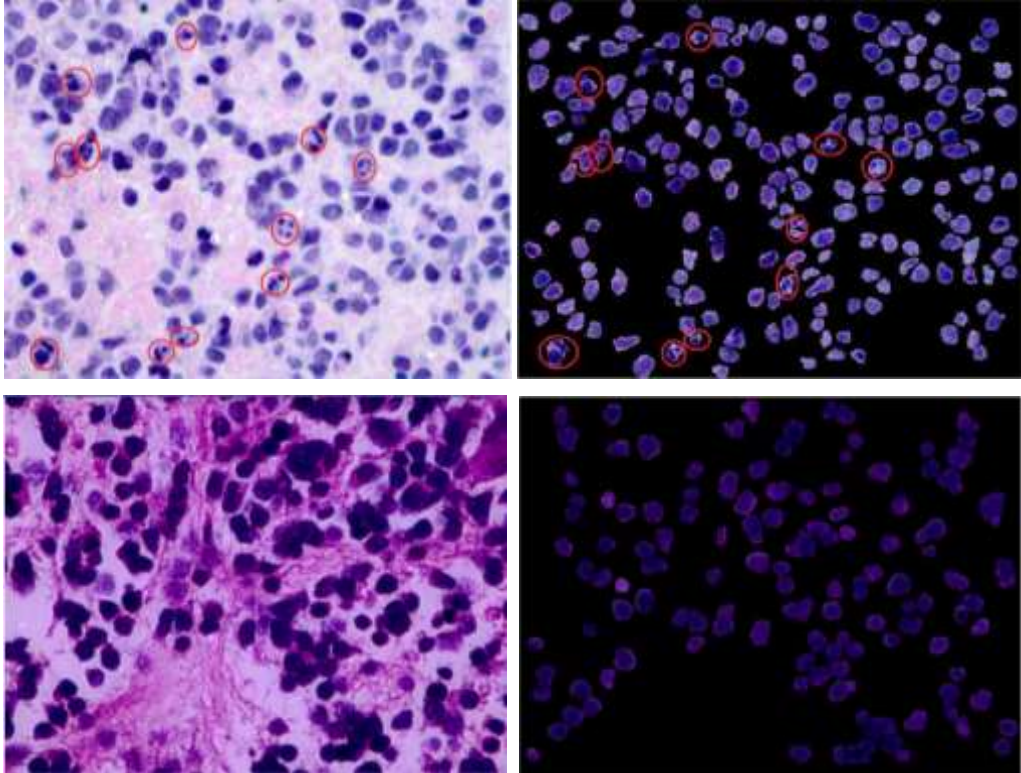


Figure 5.52 The undifferentiated type images to be used in testing phase. The mitosis and karyorrhexis cells are indicated with red circles (left), the images obtained after the application of previously described operations to the original images (right).

The test images to be used for ANNs or ensemble classifiers for MK index are shown in Figure 5.52. Similarly, the test images to be used for ANNs or ensemble classifiers for the grade of differentiation subtype are shown in Figure 5.53.

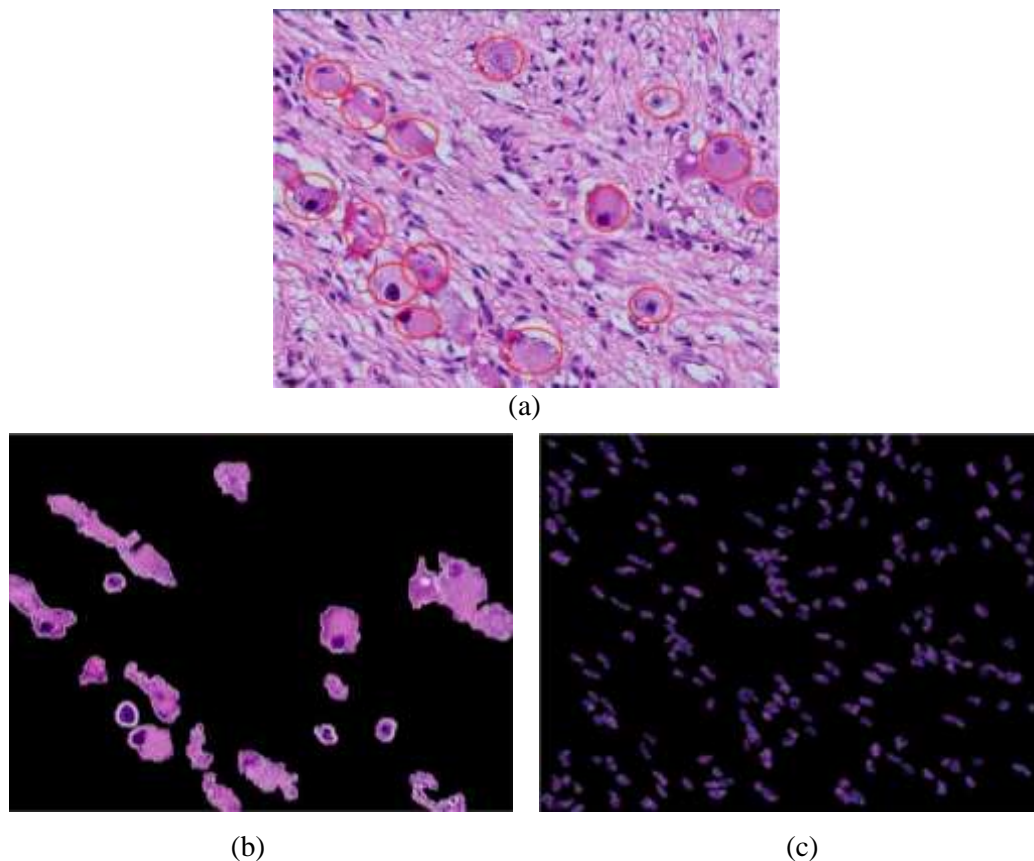


Figure 5.53 (a) The differentiated type image to be used in testing phase. The differentiated cells are indicated with red circles, (b), (c) The images obtained after the application of previously described operations to the original image.

5.5.2 Artificial Neural Network

There are quite many variations for building ANNs. In our simulations, we use a feed forward back-propagation network together with different resampling methods such as cross validation, bootstrap, etc. The details of the used network are given below.

- Transfer Function : tansig
- Training Function : Scaled Conjugate Gradient (traincsg)
- Performance Function : mse
- Number of Total Features : 24
- Number of Hidden Layers : 1
- Number of Hidden Neurons: 40 (if entire feature set is used) and 20 (if PCA or feature selection methods are used)

In order to train and test the neural network, feature set and targets (labeled cells) must be obtained. The feature set is obtained by application of feature extraction algorithms together with some image processing algorithms. However, target data must be labeled on the images manually by medical experts in order to determine the cell types. The block diagram of the training section is given below (Figure 5.54).

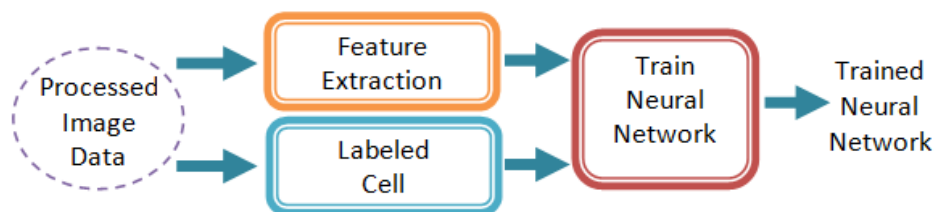


Figure 5.54 Block diagram for training the neural network.

Two different neural networks are created in the classification stage; one for grade of differentiation subtype and the other for MK index subtype. In each network, one image is left out for testing purpose and the other images are used for training the network.

Different resampling methods are used in order to obtain the best performance. The steps of the used methods are explained below.

5.5.2.1 Cross Validation

In order to apply this method, the feature set is divided into subsets by using *crossvalind* command in MATLAB. Since we have used 10-fold cross validation, ten subsets are randomly obtained.

The neural network is trained ten times. Each time nine of the subsets are used for training the network and one of the subsets is used to test the network. For each training session, the single subset that is used for testing is changed. Thus, different training sets are used at each session. The final neural network is obtained in this way.

5.5.2.2 *Bootstrap*

For the application of this method, again different feature subsets are required. These subsets are obtained by randomly selecting the features from the feature set. The difference of the bootstrap from the cross validation is that the size of the subsets is determined by the user without any limitation whereas the size of the subsets in cross validation method is fixed and calculated as the size of the feature set over number of folds.

We have obtained 20 feature subsets each including 500 elements for training the network and the network is trained 20 times by using one subset at each time. By this way the final network is obtained in order to be used for testing purposes.

5.5.2.3 *Random Selection*

We have two classes for both grade of differentiation and MK index subtypes. For both of these subtypes, the number of tumor cells (labeled with zero) are excessive when compared to the other class. Since one class dominates the other, during the training, one cell type is overtrained while the other class is mostly ignored.

In order to prevent this problem, training with fewer samples is realized. 50 random samples are selected from the dominant class and all the samples of the weak class are used. By this way, the weights of both classes are almost equalized and overtraining problem is solved to a great degree.

5.5.2.4 *Increasing the Number of Weak Class*

Another method we have applied to equalize the weights of both classes is to increase the number of differentiated and MK index cells (the cells labeled with one). Each of the features from the weak class is replicated twenty times, and the entire feature set of tumor cells are added to these features. By this way, the total feature set is obtained and the training is performed by using this set. Thus, the weights of both classes are equalized somehow.

5.5.3 Ensemble Methods

In this study, ensemble methods which are alternative methods to neural networks are also used. The reason for this is that ensemble is a technique for combining many weak learners in an attempt to produce a strong learner. Because we have a weak learner, we have decided to try this method, as well.

The MATLAB command *fitensemble* is used with different parameters and a classifier object is obtained. By using *predict* command in MATLAB, the test of trained classifier is performed.

Two ensemble methods are used in this study; AdaBoost and RobustBoost together with discriminant and tree methods.

5.5.3.1 Adaptive Boosting

Adaptive Boosting (AdaBoost), is a machine learning algorithm, formulated by Freund, Y. & Schapire, E. R. (1995). One of the main ideas of the algorithm is to maintain a distribution or set of weights over the training set. Initially, all weights are set equally, but on each round, the weights of incorrectly classified examples are increased so that the weak learner is forced to focus on hard examples in the training set. By this way, AdaBoost constructs a composite classifier by sequentially training classifiers, while putting more and more emphasis on certain patterns.

There are many extensions of AdaBoost algorithm for classification of two and multiple classes. In this study, we have used AdaBoostM1 which is used for binary classification. We have used AdaBoost method with both tree and discriminant learners. Number of learning cycles is taken as 40.

5.5.3.2 Robust Boosting

Robust Boosting (RobustBoost) presented by Freund, Y. (2009) is a new boosting algorithm which is significantly more robust against label noise than AdaBoost.

We have also tested RobustBoost algorithm with our data set. This time only discriminant learner is tested, because this method does not allow the use of tree learner. The number of learning cycles is again taken as 40.

5.5.4 Dimensionality Reduction

All the above explained methods are used together with dimension reduction methods in order to get a better performance. For this purpose, principal component analysis (PCA), which is a feature extraction method, and a feature selection method are used. All the explained methods in the previous sections for neural networks and ensemble methods are tested with three options; normal training, which is the training type without any dimensionality reduction, PCA, and feature selection.

5.5.4.1 Normal Training

This is the training type we used without any dimensionality reduction. All the extracted 24 features are used with this training option.

5.5.4.2 Principal Component Analysis (PCA)

In PCA, the feature set is mapped into another feature set whose size is smaller than the original one. For our data set, 24 features are reduced to 9 by using this method and the networks are trained with this reduced set. Neural networks with different resampling algorithms and ensemble methods are trained together with PCA option.

5.5.4.3 Feature Selection

Feature selection is used to select some of the features inside the entire feature set and discard the other less relevant features. This selection is performed with an algorithm called *sequentialfs* in MATLAB. This time, features are not mapped into a new feature set. Instead, the features stay the same; only the selected features are used for training and testing the network.

As already mentioned, each of the neural networks and the ensemble classifiers described in Sections 5.5.2 and 5.5.3 are trained with three different dimensionality reduction techniques. For example, three neural networks are created for the cross validation technique; cross validation with entire feature set, cross validation with PCA, and cross validation with feature selection. Besides, each of the neural networks is trained 20 times and the average of these results is calculated. This is because the training results vary at each training session. For the training of the ensemble classifiers, the averaging is not necessary, because for these classifiers, the results of the training remain the same for each training.

In the beginning of this section, it has been mentioned that all the images except one image from grade of differentiation subtype and one image from MK index subtype are used for training. The images reserved for testing are used to test each of the trained networks and ensemble classifiers.

In the Chapter Six, the tables including the training and test results for each neural network and ensemble classifier described in this section are given together with some added comments.

5.6 Graphical User Interface (GUI)

In this section, the Graphical User Interface (GUI) designed for a user-friendly control of the developed algorithm is explained.

In Figure 5.55, the designed GUI is shown. The analysis is performed in two stages in our GUI. First, the image taken under 20x magnification is selected by using the “SELECT IMAGE” button on the left hand side and then the “RUN” button is pressed. This button makes the “NPDA” algorithm to run. The output of this algorithm is printed under the image on the left of the screen. For example, the output of the algorithm is obtained as “undifferentiated type” for the image given in Figure 5.55. As a result of the first algorithm, the user is guided to select the 100x zoomed image.



Figure 5.55 The screenshot of GUI for mitosis karyorrhexis (MK) subtype.

In the second stage, according to the determined image type with respect to the magnification, the new image must be selected by using the second “SELECT IMAGE” button on the right hand side. The selected image is shown in the middle. After selecting the image, the “RUN” button next to it must be pressed. This “RUN” button makes “CDDI” or “CDUI” algorithm to run based on the magnification of the selected image. The output image obtained from the algorithm is displayed on the right hand side. Under this image, the number of detected cells, the MK index, and the grade of differentiation rate are printed. In the sample image given in Figure 5.55, 191 cells are detected and the MK index is determined as 7.2917%. Besides, the level of MK index is determined as “High MKI”.

The other GUI screenshots for different input images are shown in Figures 5.56 and 5.57.



Figure 5.56 The screenshot of GUI for undifferentiated subtype.



Figure 5.57 The screenshot of GUI for differentiated subtype.

CHAPTER SIX

RESULTS

In this chapter, the results of ANNs and ensemble methods employed for the analysis of MK index subtype and grade of differentiation subtype are given.

6.1 The Results of MK Index Subtype

The trained neural networks and ensemble classifiers are tested with the images given in Figure 6.1. These images do not exist in the training set because they are only used for testing purpose. The neural network results for the image given in Figure 6.1 (a) are shown in Table 6.1. All the values in the tables are obtained by averaging 20 different training and test results.

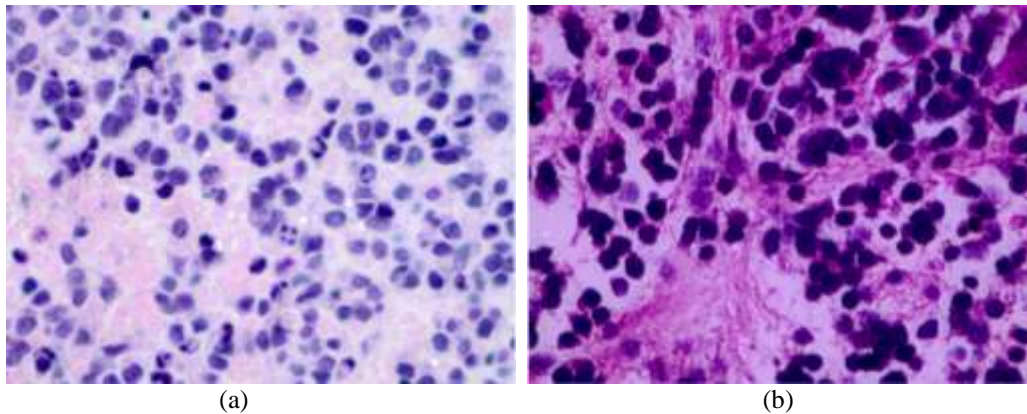


Figure 6.1 The images for analysis of MK index subtype; (a) the test image called Image_31u, (b) the test image called Image_27du.

There are 1298 samples from Class 1 (tumor cells) and 14 samples from Class 2 (mitosis and karyorrhexis cells) in the training set. The test image includes 179 cells from Class 1 and 11 cells from Class 2.

For the feature selection dimension reduction technique, six features are selected and used; correlation, cluster prominence, cluster shade, energy, maximum probability, and information measurement of correlation 2.

When we observe Table 6.1, it is seen that the test performance of MK cells (Class 2) varies between 2.73% and 80.45% for different methods. The test performance of tumor cells (Class 1) differs between 87.23% and 99.69%. The correct classification rate of tumor cells is very high, but this rate is not sufficient for mitosis and karyorrhexis cells for some cases. It can be argued that the best results for mitosis and karyorrhexis cells are obtained at random selection case. Since we decrease the number of tumor cells in this method by random selection, the classification performance of tumor cells decreases, and this causes more than 10% of tumor cells to be misclassified as mitosis and karyorrhexis cells.

Table 6.1 The classification performance results for the MK subtype using neural networks.

Dimension reduction	Resampling Techniques	Test Results for Test Image (Image_31u)						
		Class 1		Class 2		% Class 1	% Class 2	% General
		TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	Perfor.
-	-	176,7	2,4	2,5	8,6	98,69	22,27	94,26
PCA	-	178,1	0,9	0,8	10,3	99,50	6,82	94,13
Feature Selection	-	178,5	0,6	0,3	10,7	99,69	2,73	94,08
-	Cross Validation	173,2	5,8	6,0	5,1	96,79	54,09	94,31
PCA	Cross Validation	174,0	5,0	5,6	5,5	97,21	50,45	94,50
Feature Selection	Cross Validation	172,5	6,5	6,3	4,7	96,37	57,27	94,11
-	Bootstrap	172,5	6,6	6,4	4,6	96,34	58,18	94,13
PCA	Bootstrap	175,0	4,1	5,0	6,0	97,74	45,45	94,71
Feature Selection	Bootstrap	174,3	4,7	5,1	5,9	97,37	46,36	94,42
-	Random Selection	159,4	19,7	8,9	2,2	89,02	80,45	88,53
PCA	Random Selection	156,2	22,9	8,8	2,2	87,23	80,00	86,82
Feature Selection	Random Selection	160,0	19,1	8,4	2,6	89,36	76,36	88,61
-	Increase Cl2 Samples	171,0	8,1	6,3	4,8	95,50	56,82	93,26
PCA	Increase Cl2 Samples	174,8	4,2	5,5	5,5	97,67	50,00	94,91
Feature Selection	Increase Cl2 Samples	169,4	9,6	5,4	5,6	94,64	49,09	92,00

Possibly more important results to be observed are the percentages of MK indices which are shown in Table 6.2. For pathologists, the classification is not enough; they also use the MK index in order to determine the stage of the disease. The MK index is calculated as the number of mitosis and karyorrhexis cells over the number of all tumor cells (mitosis and karyorrhexis cells plus tumor cells). It is seen that the MK index is obtained around 15% for the random selection algorithm which was mentioned as successful in the above paragraph. The correct MK index is approximately 5.8%, hence the random selection algorithm has produced the worst results in terms of percentage of MK index. Cross validation and bootstrap methods give better results when compared to the other methods.

Table 6.2 The percentage of MK index results for the MK subtype using neural networks.

Dimension reduction	Resampling Techniques	MK INDEX PERCENT (For Test Image)					
		Real Data			Simulated Data		
		# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	-	179,00	11,00	5,789	185,20	4,80	2,526
PCA	-	179,00	11,00	5,789	188,35	1,65	0,868
Feature Selection	-	179,00	11,00	5,789	189,15	0,85	0,447
-	Cross Validation	178,95	11,00	5,791	178,25	11,70	6,160
PCA	Cross Validation	179,00	11,00	5,789	179,45	10,55	5,553
Feature Selection	Cross Validation	179,00	11,00	5,789	177,20	12,80	6,737
-	Bootstrap	179,00	11,00	5,789	177,05	12,95	6,816
PCA	Bootstrap	179,00	11,00	5,789	180,95	9,05	4,763
Feature Selection	Bootstrap	179,00	11,00	5,789	180,20	9,80	5,158
-	Random Selection	179,00	11,00	5,789	161,50	28,50	15,000
PCA	Random Selection	179,00	11,00	5,789	158,35	31,65	16,658
Feature Selection	Random Selection	179,00	11,00	5,789	162,55	27,45	14,447
-	Increase Cl2 Samples	179,00	11,00	5,789	175,70	14,30	7,526
PCA	Increase Cl2 Samples	179,00	11,00	5,789	180,33	9,67	5,088
Feature Selection	Increase Cl2 Samples	179,00	11,00	5,789	175,00	15,00	7,895

In Table 6.3, the classification performance results for the image given in Figure 6.1 (a) by using ensemble classifiers are shown. It is not necessary to train the ensemble classifier 20 times, because it always gives the same results. It is seen that the classification performance for tumor cells changes between 94.97% and 99.44%. For mitosis and karyorrhexis cells the classification performance varies between 27.27% and 72.73%. The general performance is around 95%. The RobustBoost discriminant and AdaBoost discriminant methods produce acceptable results for both classes.

Table 6.3 The classification performance results for the MK subtype using ensemble methods.

Dimension Reduction	Ensemble Method	Test Results for Test Image (Image_31u)						
		Class 1		Class 2		% Class 1	% Class 2	%
		TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	General
-	Adaboost Discriminant	170,0	9,0	8,0	3,0	94,97	72,73	93,68
PCA	Adaboost Discriminant	175,0	4,0	7,0	4,0	97,77	63,64	95,79
Feature Selection	Adaboost Discriminant	178,0	1,0	3,0	8,0	99,44	27,27	95,26
-	Adaboost Tree	174,0	5,0	5,0	6,0	97,21	45,45	94,21
PCA	Adaboost Tree	175,0	4,0	6,0	5,0	97,77	54,55	95,26
Feature Selection	Adaboost Tree	171,0	8,0	5,0	6,0	95,53	45,45	92,63
-	Robustboost Discriminant	171,0	8,0	8,0	3,0	95,53	72,73	94,21
PCA	Robustboost Discriminant	171,0	8,0	7,0	4,0	95,53	63,64	93,68
Feature Selection	Robustboost Discriminant	174,0	5,0	5,0	6,0	97,21	45,45	94,21

In Table 6.4, the percentages of MK indices for ensemble methods are given. Judging by the obtained values, it follows that the results are generally acceptable.

The closest result to the true value is obtained when the AdaBoost discriminant method is utilized with PCA dimension reduction technique.

Table 6.4 The percentage of MK index results for the MK subtype using ensemble methods.

Dimension Reduction	Ensemble Method	MK INDEX PERCENT (For Test Image)					
		Real Data			Simulated Data		
		# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	Adaboost Discriminant	179,00	11,00	5,789	173,00	17,00	8,947
PCA	Adaboost Discriminant	179,00	11,00	5,789	179,00	11,00	5,789
Feature Selection	Adaboost Discriminant	179,00	11,00	5,789	186,00	4,00	2,105
-	Adaboost Tree	179,00	11,00	5,789	180,00	10,00	5,263
PCA	Adaboost Tree	179,00	11,00	5,789	180,00	10,00	5,263
Feature Selection	Adaboost Tree	179,00	11,00	5,789	177,00	13,00	6,842
-	Robustboost Discriminant	179,00	11,00	5,789	174,00	16,00	8,421
PCA	Robustboost Discriminant	179,00	11,00	5,789	175,00	15,00	7,895
Feature Selection	Robustboost Discriminant	179,00	11,00	5,789	180,00	10,00	5,263

For the image given in Figure 6.1 (b), which is called Image_27du, the neural network results are given in Table 6.5. As seen from the table, the classification percentages are quite high for this image, because the colors are more distinctive in this image. The test image has no mitosis and karyorrhesis cells and the classification performance is 100% or around for most of the methods.

Table 6.5 The classification performance results for the MK subtype using neural networks.

Dimension reduction	Resampling Techniques	Test Results for Test Image (Image_27du)						
		Class 1		Class 2		% Class 1	% Class 2	% General
		TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	Perfor.
-	-	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	-	115,0	0,0	0,0	0,0	100,00	100,00	100,00
Feature Selection	-	115,0	0,0	0,0	0,0	100,00	100,00	100,00
-	Cross Validation	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	Cross Validation	115,0	0,0	0,0	0,0	100,00	100,00	100,00
Feature Selection	Cross Validation	115,0	0,0	0,0	0,0	100,00	100,00	100,00
-	Bootstrap	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	Bootstrap	115,0	0,0	0,0	0,0	100,00	100,00	100,00
Feature Selection	Bootstrap	115,0	0,0	0,0	0,0	100,00	100,00	100,00
-	Random Selection	114,7	0,4	0,0	0,0	99,70	100,00	99,70
PCA	Random Selection	99,9	15,2	0,0	0,0	86,83	100,00	86,83
Feature Selection	Random Selection	114,6	0,4	0,0	0,0	99,65	100,00	99,65
-	Increase Cl2 Samples	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	Increase Cl2 Samples	113,3	1,7	0,0	0,0	98,55	100,00	98,55
Feature Selection	Increase Cl2 Samples	114,6	0,4	0,0	0,0	99,65	100,00	99,65

Table 6.6 shows the percentage results of MK index for each method. However, for this image this analysis is not very meaningful, because there are no mitosis and karyorrhesis cells in this image. The random selection method used with PCA

algorithm has produced the worst result; other methods produced results having satisfactorily high accuracy.

Table 6.6 The percentage of MK index results for the MK subtype using neural networks.

Dimension reduction	Resampling Techniques	MK INDEX PERCENT (For Test Image)					
		Real Data			Simulated Data		
		# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	-	115,00	0,00	0,000	115,00	0,00	0,000
PCA	-	115,00	0,00	0,000	115,00	0,00	0,000
Feature Selection	-	115,00	0,00	0,000	115,00	0,00	0,000
-	Cross Validation	115,00	0,00	0,000	115,00	0,00	0,000
PCA	Cross Validation	115,00	0,00	0,000	115,00	0,00	0,000
Feature Selection	Cross Validation	115,00	0,00	0,000	115,00	0,00	0,000
-	Bootstrap	115,00	0,00	0,000	115,00	0,00	0,000
PCA	Bootstrap	115,00	0,00	0,000	115,00	0,00	0,000
Feature Selection	Bootstrap	115,00	0,00	0,000	115,00	0,00	0,000
-	Random Selection	115,00	0,00	0,000	114,65	0,35	0,304
PCA	Random Selection	115,00	0,00	0,000	99,85	15,15	13,174
Feature Selection	Random Selection	115,00	0,00	0,000	114,60	0,40	0,348
-	Increase Cl2 Samples	115,00	0,00	0,000	115,00	0,00	0,000
PCA	Increase Cl2 Samples	115,00	0,00	0,000	113,33	1,67	1,449
Feature Selection	Increase Cl2 Samples	115,00	0,00	0,000	114,60	0,40	0,348

Table 6.7 The classification performance results for the MK subtype using ensemble methods.

Dimension Reduction	Ensemble Method	Test Results for Test Image (Image_27du)						
		Class 1		Class 2		% Class 1	% Class 2	%
		TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	General
-	Adaboost Discriminant	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	Adaboost Discriminant	115,0	0,0	0,0	0,0	100,00	100,00	100,00
Feature Selection	Adaboost Discriminant	115,0	0,0	0,0	0,0	100,00	100,00	100,00
-	Adaboost Tree	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	Adaboost Tree	115,0	0,0	0,0	0,0	100,00	100,00	100,00
Feature Selection	Adaboost Tree	115,0	0,0	0,0	0,0	100,00	100,00	100,00
-	Robustboost Discriminant	115,0	0,0	0,0	0,0	100,00	100,00	100,00
PCA	Robustboost Discriminant	115,0	0,0	0,0	0,0	100,00	100,00	100,00
Feature Selection	Robustboost Discriminant	115,0	0,0	0,0	0,0	100,00	100,00	100,00

Table 6.8 The percentage of MK index results for the MK subtype using ensemble methods.

Dimension Reduction	Ensemble Method	MK INDEX PERCENT (For Test Image)					
		Real Data			Simulated Data		
		# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	Adaboost Discriminant	115,00	0,00	0,000	115,00	0,00	0,000
PCA	Adaboost Discriminant	115,00	0,00	0,000	115,00	0,00	0,000
Feature Selection	Adaboost Discriminant	115,00	0,00	0,000	115,00	0,00	0,000
-	Adaboost Tree	115,00	0,00	0,000	115,00	0,00	0,000
PCA	Adaboost Tree	115,00	0,00	0,000	115,00	0,00	0,000
Feature Selection	Adaboost Tree	115,00	0,00	0,000	115,00	0,00	0,000
-	Robustboost Discriminant	115,00	0,00	0,000	115,00	0,00	0,000
PCA	Robustboost Discriminant	115,00	0,00	0,000	115,00	0,00	0,000
Feature Selection	Robustboost Discriminant	115,00	0,00	0,000	115,00	0,00	0,000

Table 6.7 shows the results for Image_27du in Figure 6.1 (b) by using ensemble methods. All the cells are correctly classified by all methods. In Table 6.8, the calculated percentage of MK index for each classifier is given. The results are obtained with 100% accuracy for all methods.

6.2 The Results of Grade of Differentiation Subtype

The trained neural networks and ensemble classifiers for the grade of differentiation subtype are tested with the image given in Figure 6.2. This image includes 199 cells from Class 1 (tumor cells) and 14 cells from Class 2 (differentiated cells). There are 1298 cells from Class 1 (tumor cells) and 14 cells from Class 2 (mitosis and karyorrhexis cells) in the training set.

For the feature selection dimension reduction technique, five features are selected and used; autocorrelation, entropy, sum variance, difference variance, and difference entropy.

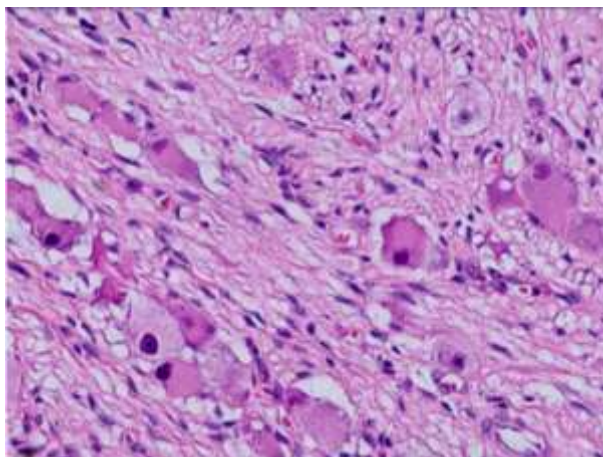


Figure 6.2 The image (Image_57_1) for analysis of grade of differentiation subtype.

In Table 6.9, the test results for the trained neural networks with different resampling techniques are given. As seen from the table, the classification rates are quite high. Because of that, not all methods are tested for this subtype. Again, for the

neural networks the training and test procedures are performed 20 times and the averaged results are given in the table.

Table 6.9 The classification performance results for the grade of differentiation subtype using neural networks.

Dimension reduction	Resampling Techniques	Test Results for Test Image (Image_27d)						
		Class 1		Class 2		% Class 1	% Class 2	% General
		TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	
-	-	196,95	2,05	13,35	0,65	98,97	95,36	98,73
PCA	-	196,95	2,05	12,80	1,20	98,97	91,43	98,47
Feature Selection	-	197,65	1,35	11,60	2,40	99,32	82,86	98,24
-	Cross Validation	197,00	2,00	13,00	1,00	98,99	92,86	98,59
PCA	Cross Validation	196,15	2,85	13,00	1,00	98,57	92,86	98,19
Feature Selection	Cross Validation	196,80	2,20	12,40	1,60	98,89	88,57	98,22
-	Bootstrap	196,90	2,10	13,10	0,90	98,94	93,57	98,59
PCA	Bootstrap	197,15	1,85	13,00	1,00	99,07	92,86	98,66
Feature Selection	Bootstrap	197,85	1,15	12,15	1,85	99,42	86,79	98,59

In Table 6.10, the percentage of MK index for each method is given. The general performance is around 98%. Since the classification rate is quite high, the calculated MK index percentage is very close to the true value.

Table 6.10 The percentage of grade of differentiation results for the grade of differentiation subtype using neural networks.

Dimension Reduction	Resampling Techniques	GRADE OF DIFFERENTIATION PERCENT (For Test Image)					
		Real Data			Simulated Data		
		# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	-	199	14	6,573	197,60	15,40	7,230
PCA	-	199	14	6,573	198,15	14,85	6,972
Feature Selection	-	199	14	6,573	200,05	12,95	6,080
-	Cross Validation	199	14	6,573	198,00	15,00	7,042
PCA	Cross Validation	199	14	6,573	197,15	15,85	7,441
Feature Selection	Cross Validation	199	14	6,573	198,40	14,60	6,854
-	Bootstrap	199	14	6,573	197,80	15,20	7,136
PCA	Bootstrap	199	14	6,573	198,15	14,85	6,972
Feature Selection	Bootstrap	199	14	6,573	199,70	13,30	6,244

The test results using the ensemble classifiers with the same image are given in Table 6.11. The results of some methods are a little better when compared to neural networks. The worst performance is obtained as 97.18% and the best performance is obtained as 99.53%.

Table 6.11 The classification performance results for the grade of differentiation subtype using ensemble methods.

Dimension Reduction	Ensemble Method	Test Results for Test Image (Image_27d)						
		Class 1		Class 2		% Class 1	% Class 2	%
		TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	General
-	Adaboost Discriminant	198,00	1,00	13,00	1,00	99,50	92,86	99,06
PCA	Adaboost Discriminant	198,00	1,00	14,00	0,00	99,50	100,00	99,53
Feature Selection	Adaboost Discriminant	197,00	2,00	10,00	4,00	98,99	71,43	97,18
-	Adaboost Tree	198,00	1,00	14,00	0,00	99,50	100,00	99,53
PCA	Adaboost Tree	197,00	2,00	13,00	1,00	98,99	92,86	98,59
Feature Selection	Adaboost Tree	199,00	0,00	10,00	4,00	100,00	71,43	98,12
-	RobustBoost Discriminant	198,00	1,00	13,00	1,00	99,50	92,86	99,06
PCA	RobustBoost Discriminant	198,00	1,00	13,00	1,00	99,50	92,86	99,06
Feature Selection	RobustBoost Discriminant	196,00	3,00	13,00	1,00	98,49	92,86	98,12

The percentage of MK index for each classifier is given in Table 6.12. As seen from the table, AdaBoost discriminant method with no dimension reduction and RobustBoost discriminant method with no dimension reduction and PCA give the best performance.

Table 6.12 The percentage of grade of differentiation results for the grade of differentiation subtype using ensemble methods.

Dimension Reduction	Ensemble Method	GRADE OF DIFFERENTIATION PERCENT (For Test Image)					
		Real Data			Simulated Data		
		# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	Adaboost Discriminant	199	14	6,573	199,00	14,00	6,573
PCA	Adaboost Discriminant	199	14	6,573	198,00	15,00	7,042
Feature Selection	Adaboost Discriminant	199	14	6,573	201,00	12,00	5,634
-	Adaboost Tree	199	14	6,573	198,00	15,00	7,042
PCA	Adaboost Tree	199	14	6,573	198,00	15,00	7,042
Feature Selection	Adaboost Tree	199	14	6,573	203,00	10,00	4,695
-	RobustBoost Discriminant	199	14	6,573	199,00	14,00	6,573
PCA	RobustBoost Discriminant	199	14	6,573	199,00	14,00	6,573
Feature Selection	RobustBoost Discriminant	199	14	6,573	197,00	16,00	7,512

6.3 The Results of Using Different Number of Neurons at ANNs

One of the most important parameters of ANNs is the number of hidden neurons. In this section, we have performed some tests in order to see the effect of the number of hidden neurons. Tables 6.13 through 6.16 show the results for the MK index subtype by using cross validation and bootstrap methods. From these results, it follows that the number of neurons directly affects the performance. If the number of hidden neurons is fewer than the number of features, the classification performance

decreases. It is seen that using more hidden neurons than the number of features increases the performance. However, using excessively many neurons decreases the performance. This may be attributed to the overfitting problem.

Table 6.13 The classification performance results for MK index subtype using neural networks with different number of neurons.

Dimension Reduction	Resampling Techniques	# of Neurons	Test Results for Test Image (Image_31u)						
			Class 1		Class 2		% Class 1	% Class 2	% General
			TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	Perfor.
-	Cross Validation	5	172,3	6,7	5,2	5,8	96,26	47,27	93,42
-	Cross Validation	10	173,6	5,4	4,4	6,6	96,98	40,00	93,68
-	Cross Validation	24	171,8	7,2	5,8	5,2	95,98	52,73	93,47
-	Cross Validation	40	173,2	5,8	6,0	5,1	96,79	54,09	94,31
-	Cross Validation	60	174,0	5,0	4,6	6,4	97,21	41,82	94,00

Table 6.14 The percentage of grade of differentiation results for MK index subtype using neural networks with different number of neurons.

Dimension Reduction	Resampling Techniques	# of Neurons	MK INDEX PERCENT (For Test Image)					
			Real Data			Simulated Data		
			# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	Cross Validation	5	179,00	11,00	5,789	178,10	11,90	6,263
-	Cross Validation	10	179,00	11,00	5,789	180,20	9,80	5,158
-	Cross Validation	24	179,00	11,00	5,789	177,00	13,00	6,842
-	Cross Validation	40	179,00	11,00	5,789	178,25	11,70	6,160
-	Cross Validation	60	179,00	11,00	5,789	180,40	9,60	5,053

Table 6.15 The classification performance results for MK index subtype using neural networks with different number of neurons and with feature selection.

Dimension Reduction	Resampling Techniques	# of Neurons	Test Results for Test Image (Image_31u)						
			Class 1		Class 2		% Class 1	% Class 2	% General
			TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	Perfor.
Feature selection	Bootstrap	2	176,9	2,1	2,2	8,8	98,83	20,00	94,26
Feature selection	Bootstrap	6	175,3	3,7	2,8	8,2	97,93	25,45	93,74
Feature selection	Bootstrap	10	176,1	2,9	2,6	8,4	98,38	23,64	94,05
Feature selection	Bootstrap	40	174,3	4,7	5,1	5,9	97,37	46,36	94,42
Feature selection	Bootstrap	60	176,9	2,1	3,2	7,8	98,83	29,09	94,79

* After feature selection, the number of features is 6.

Table 6.16 The percentage of grade of differentiation results for MK index subtype using neural networks with different number of neurons and with feature selection.

Dimension Reduction	Resampling Techniques	# of Neurons	MK INDEX PERCENT (For Test Image)					
			Real Data			Simulated Data		
			# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
Feature selection	Bootstrap	2	179,00	11,00	5,789	185,70	4,30	2,263
Feature selection	Bootstrap	6	179,00	11,00	5,789	183,50	6,50	3,421
Feature selection	Bootstrap	10	179,00	11,00	5,789	184,50	5,50	2,895
Feature selection	Bootstrap	40	179,00	11,00	5,789	180,20	9,80	5,158
Feature selection	Bootstrap	60	179,00	11,00	5,789	184,70	5,30	2,789

* After feature selection, the number of features is 6.

Table 6.17 The classification performance results for the grade of differentiation subtype using neural networks with different number of neurons.

Dimension Reduction	Resampling Techniques	# of Neurons	Test Results for Test Image (Image_51d)						
			Class 1		Class 2		% Class 1	% Class 2	% General
			TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	Perfor.
-	Cross Validation	5	196,9	2,1	13,0	1,0	98,94	92,86	98,54
-	Cross Validation	10	196,8	2,2	13,0	1,0	98,89	92,86	98,50
-	Cross Validation	24	196,8	2,2	13,1	0,9	98,89	93,57	98,54
-	Cross Validation	40	197,0	2,0	13,0	1,0	98,99	92,86	98,59
-	Cross Validation	60	197,0	2,0	13,0	1,0	98,99	92,86	98,59

Table 6.18 The percentage of grade of differentiation results for the grade of differentiation subtype using neural networks with different number of neurons.

Dimension Reduction	Resampling Techniques	# of Neurons	GRADE OF DIFFERENTIATION PERCENT (For Test Image)					
			Real Data			Simulated Data		
			# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
-	Cross Validation	5	199,00	14,00	6,573	197,90	15,10	7,089
-	Cross Validation	10	199,00	14,00	6,573	197,80	15,20	7,136
-	Cross Validation	24	199,00	14,00	6,573	197,70	15,30	7,183
-	Cross Validation	40	199,00	14,00	6,573	198,00	15,00	7,042
-	Cross Validation	60	199,00	14,00	6,573	198,00	15,00	7,042

Table 6.19 The classification performance results for the grade of differentiation subtype using neural networks with different number of neurons and with feature selection.

Dimension Reduction	Resampling Techniques	# of Neurons	Test Results for Test Image (Image_51d)						
			Class 1		Class 2		% Class 1	% Class 2	% General
			TRUE	FALSE	TRUE	FALSE	Perfor.	Perfor.	Perfor.
Feature selection	Bootstrap	2	197,5	1,5	12,0	2,0	99,25	85,71	98,36
Feature selection	Bootstrap	5	197,3	1,7	11,6	2,4	99,15	82,86	98,08
Feature selection	Bootstrap	10	197,6	1,4	11,7	2,3	99,30	83,57	98,26
Feature selection	Bootstrap	20	197,9	1,2	12,2	1,9	99,42	86,79	98,59
Feature selection	Bootstrap	60	197,2	1,8	12,0	2,0	99,10	85,71	98,22

* After feature selection, the number of features is 5.

Table 6.20 The percentage of grade of differentiation results for the grade of differentiation subtype using neural networks with different number of neurons and with feature selection.

Dimension Reduction	Resampling Techniques	# of Neurons	GRADE OF DIFFERENTIATED PERCENT (For Test Image)					
			Real Data			Simulated Data		
			# of Class1	# of Class 2	%	# of Class1	# of Class 2	%
Feature selection	Bootstrap	2	199,00	14,00	6,573	199,50	13,50	6,338
Feature selection	Bootstrap	5	199,00	14,00	6,573	199,70	13,30	6,244
Feature selection	Bootstrap	10	199,00	14,00	6,573	199,90	13,10	6,150
Feature selection	Bootstrap	20	199,00	14,00	6,573	199,70	13,30	6,244
Feature selection	Bootstrap	60	199,00	14,00	6,573	199,20	13,80	6,479

* After feature selection, the number of features is 5.

The same tests are also performed for grade of differentiation subtype. Tables 6.17 through 6.20 show the results. Since the classification is easier for this subtype, the effect of increasing the number of hidden neurons is not so evident. Nevertheless, similar comments can also be made for this subtype as well.

CHAPTER SEVEN

CONCLUSIONS

Neuroblastoma (NB) is a cancer of nerve cell origin commonly affecting infants and children. As described earlier, histopathological examinations performed by expert pathologists are required to characterize the histology for further treatment planning of the tumor. Throughout this thesis work, we have developed an algorithm by using image processing and classification techniques. The purpose of this algorithm is to develop a system which decreases the decision variations to the lowest level by simplifying the diagnosis for pathologists as much as possible.

Basically, we have used images with different magnification ratios, belonging to the tissue samples of NB patients. After the application of various image processing techniques, the feature matrices are created by using the extracted region and texture features. Since the classification problem is difficult and the color distribution of the cells creates a texture, we have used Haralick texture features. These feature matrices are used for the classification of the cells. The classification results are given in Chapter Six.

Two kinds of results are given for each subtype in Chapter Six; the first one is the classification performance results and the other one is the percentage of MK index results. The latter one is more important for pathologists, because they use this information for determining the stage of the disease. We have performed the tests with different options; various resampling methods and dimension reduction techniques are employed in order to obtain the best performance.

For the MK index subtype, the best results are obtained employing the cross validation and bootstrap resampling methods with ANNs. AdaBoost Discriminant method has produced acceptable results when ensemble methods are utilized. The classification for this subtype is quite hard, because the number of mitosis and karyorrhexis cells is not sufficient as compared to the number of tumor cells. Besides, differentiating the cells is very difficult even with visual inspection.

Nevertheless, the employed methods have produced acceptable results for this subtype.

For the grade of differentiation subtype, the classification is easier when compared to MK index subtype. Because, for this subtype, distinction of the classes is more evident. Hence, the classification performance is better than the MK index subtype. The ensemble methods have produced somewhat better results when compared to ANNs, but generally the accuracy performance for both types are over 98%.

We have also applied different number of hidden neurons for ANNs in order to observe its effect. It is seen from the results that it is necessary to use more neurons than the number of features. If the number of hidden neurons is insufficient when compared to the number of features, the classification performance decreases. Besides, using excessively many hidden neurons also decreases the classification performance.

In summary, during this thesis study, we have developed an algorithm to be used as an auxiliary tool for determining the stage of NB disease for pathologists. We have also tested the classification results of the algorithm by using different machine learning algorithms with different options. The results are generally acceptable and useful, although further research is necessary especially for the MK index subtype. Using more mitosis and karyorrhexis cell samples, extracting different features, and using different pre-processing techniques on the images are some recommendations that can be used for improving the classification performance.

REFERENCES

- Boland, M. V. (1999). *Haralick texture features*. Retrieved July 23, 2012, from http://murphylab.web.cmu.edu/publications/boland/boland_node26.html.
- Deserno, T. M. (2011). *Biomedical image processing*. Berlin: Springer-Verlag.
- Dougherty, G. (2009). *Digital image processing for medical applications*. UK: Cambridge University Press.
- Duda, R. O., Hart, P. E. & Stork, D. G. (2000). *Pattern classification* (2nd ed.). NY: Wiley-Interscience.
- Freund, Y. & Schapire, R. E. (1997). A decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences*, 55(1), 119–139.
- Freund, Y. (2009). *A more robust boosting algorithm*. Retrieved July 23, 2012, from <http://arxiv.org/pdf/0905.2138.pdf>.
- Gurcan, M. N., Pan, T., Shimada, H. & Saltz, J. (2006). Image analysis for neuroblastoma classification: Segmentation of cell nuclei. *IEEE International Conference on Engineering in Medicine and Biology Society*, 4844–4847.
- Jain, A. (1989). *Fundamentals of digital image processing*. Englewood Cliffs, NJ: Prentice-Hall.
- Jain, A. K., Mao, J. & Mohiuddin, K. M. (1996). Artificial neural networks: A tutorial. *IEEE Computer*, 31-44.

- Joshi, V. V., Cantor, A. B., Altshuler, A., Larkin, E. W., Neill, J. S. A., Shuster, J. J., et al. (1992). Recommendation for modification of terminology of neuroblastic tumors and prognostic significance of Shimada classification: a clinicopathologic study of 213 cases. *Cancer*, 69, 2183–2196.
- Haralick, R. M. (1979). Statistical and structural approaches to texture. *Proc. IEEE*, 67, 786-804.
- Kong, J., Sertel, O., Shimada, H., Boyer, K., Saltz, J. & Gurcan, M. N. (2007a). Computer-aided grading of neuroblastic differentiation: Multi-resolution and multi-classifier approach. *IEEE International Conference on Image Processing*, 525–528.
- Kong, J., Shimada, H., Boyer, K., Saltz, J. & Gurcan, M. N. (2007b). Image analysis for automated assessment of grade of neuroblastic differentiation. *IEEE International Symposium on Biomedical Imaging*, 61–64.
- Koschan, A. & Abidi, M. (2008). *Digital color image processing*. Hoboken, New Jersey: John Wiley & Sons.
- Meyer, F. (1994). *Topographic distance and watershed lines*. *Signal Processing*, vol. 38, 113-125.
- Neuroblastoma. (n.d.). *Royal Children's Hospital Department of Clinical Haematology –Information for Families*. Retrieved July 23, 2012, from http://www.wch.sa.gov.au/services/az/divisions/paedm/clinhaem/files/cancer_book/individualdiseases/Neuroblastoma.pdf.
- Nixon, M. S. & Aguado, A. S. (2002). *Feature extraction and image processing*. Newnes.

- Qualman, S. J., Bowen, J., Fitzgibbons, P. L., Cohn, S. L. & Shimada, H. (2005). Protocol for the examination of specimens from patients with neuroblastoma and related neuroblastic tumors. *Arch. Pathol. Lab. Med.*, 129, 874-883.
- Russ, J. C. (2011). *The image processing handbook (6th ed.)*. Boca Raton: CRC Press.
- Sertel, O., Catalyurek, U. V., Shimada, H. & Gurcan, M. N. (2009). Computer-aided prognosis of neuroblastoma: detection of mitosis and karyorrhexis cells in digitized histological images. *IEEE International Conference on Engineering in Medicine and Biology Society*, 1433-1436.
- Sertel, O., Kong, J., Lozanski, G., Shana'ah, A., Catalyurek, U., Saltz, J. & Gurcan, M. N. (2008a). Texture classification using nonlinear color quantization: Application to histopathological image analysis. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 597-600.
- Sertel, O., Kong, J., Shimada, H., Catalyurek, U. V., Saltz, J. H. & Gurcan, M. N. (2008b). Computer-aided prognosis of neuroblastoma on whole-slide images: Classification of stromal development. *Pattern Recognition*, 1093-1103.
- Shimada, H., Ambros, I. M., Dehner, L. P., Hata, J., Joshi, V. V., Roald, B., Stram, D. O., Gerbing, R. B., Lukens, J. N., Matthay, K. K. & Castleberry, R. P. (1999a). The international neuroblastoma pathology classification (the Shimada system). *Cancer*, 86, 364-372.
- Shimada, H., Ambros, I. M., Dehner, L. P., Hata, J., Joshi, V. V. & Roald, B. (1999b). Terminology and morphologic criteria of neuroblastic tumors: Recommendation by the International Neuroblastoma Pathology Committee. *Cancer*, 86, 349-363.
- The HSV Colorspace. (n.d.). Retrieved July 23, 2012, from <http://ie.technion.ac.il/CC/Gimp/node51.html>.