Weighted Opposition-Based Fuzzy Thresholding

by

Pegah Ensafi

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Systems Design Engineering

Waterloo, Ontario, Canada, 2011

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

With the rapid growth of the digital imaging, image processing techniques are widely involved in many industrial and medical applications. Image thresholding plays an essential role in image processing and computer vision applications. It has a vast domain of usage. Areas such document image analysis, scene or map processing, satellite imaging and material inspection in quality control tasks are examples of applications that employ image thresholding or segmentation to extract useful information from images.

Medical image processing is another area that has extensively used image thresholding to help the experts to better interpret digital images for a more accurate diagnosis or to plan treatment procedures.

Opposition-based computing, on the other hand, is a recently introduced model that can be employed to improve the performance of existing techniques. In this thesis, the idea of oppositional thresholding is explored to introduce new and better thresholding techniques.

A recent method, called Opposite Fuzzy Thresholding (OFT), has involved fuzzy sets with opposition idea, and based on some preliminary experiments seems to be reasonably successful in thresholding some medical images.

In this thesis, a Weighted Opposite Fuzzy Thresholding method (WOFT) will be presented that produces more accurate and reliable results compared to the parent algorithm. This claim has been verified with some experimental trials using both synthetic and real world images.

Experimental evaluations were conducted on two sets of synthetic and medical images to validate the robustness of the proposed method in improving the accuracy of the thresholding process when fuzzy and oppositional ideas are combined.

Acknowledgment

I would like to express my deepest appreciation to my supervisor, Professor Hamid R. Tizhoosh for his inspiring support, advice, and encouragement throughout this work.

I would like to acknowledge my thesis readers, Professor Otman Basir and Professor John S. Zelek for providing me with their valuable comments.

I am grateful to my manager at work, Karen Vanderkruk, for all her consideration and patience with my study.

Special and warmest thanks to my husband, Nader Fathi, who has been abiding source of love and encouragement. This work would have never been possible without his wide spread support.

Finally, I would like to extend my heart-felt thanks to my parents for their blessing, my sister, Neda, for taking a good care of my kids, and my children, Kiara and Kasra, for their tolerance while mommy was busy studying.

To Nader, Kiara, and Kasra

for their love and support.

To my late father- in-law

who never stopped sharing his wisdom and encouragement to study.

He was hoping to see this work done but unfortunately passed away just a few weeks before its completion.

I wish peace to his soul.

Table of Contents

List of Figures

List of Tables

List of Pseudo-codes

CHAPTER 1

Introduction

1.1 Objective

The development of digital imaging started back in 1960s. The main idea at that time was to improve information capturing techniques especially for scientific and military purposes. In subsequent decades, as digital technology became faster and cheaper, it replaced the old image capturing methods for many purposes. Across the world, significant efforts were directed towards providing computer algorithms to perform image processing tasks on digital images; this created a rapidly growing area of computer science and engineering called Digital Image Processing. With the introduction of fast computers and signal processors, digital image processing has become the most common form of image processing because it is not only the most resourceful and flexible method, but also the cheapest. It introduced the use of much more complex algorithms; in various fields such as feature extraction, classification, and pattern recognition, that led to reach an advanced level of accuracy and performance that was impossible by analog imaging. [Wikipedia]

The main concern of digital image processing is to extract useful information from images, with as little as possible or no human intervention. Algorithms for image

processing can be categorized in three levels [34]: 1) techniques that directly deal with raw and noisy pixels, such as de-noising and edge detection algorithms, construct the lowest level, 2) methods that extend the use of low level results for more convoluted tasks such as segmentation, and 3) high-level methods that use the extracted information from previous levels to obtain semantic meanings from the images. Handwriting recognition is an example of methods in this level.

Image thresholding and segmentation is one of the well known and widely used image processing methods. Its focus is on the task of classification of image pixels into distinct and generally disjoint classes based on their characteristics. Many thresholding algorithms take advantage of various optimality measures to define the best threshold [7]. Amongst these methods image histogram is used, globally or locally, as a basis to perform the task of thresholding [7], [8]. The primary objective of this thesis is to increase the accuracy of one of the recently introduced thresholding methods based on information derived from local histograms of the image.

1.2 Contribution

The major concern with thresholding methods is to not only consider the gray-level intensities, but also the relationship between the pixels. A large number of studies have been conducted to address this concern within different scales and point of views. In research performed by Tizhoosh et al. [32], the authors introduced opposition-based fuzzy thresholding, called OFT henceforward, and combine the concepts of fuzzy memberships and opposition-based computing to extract some local information of the image that leads to selecting a threshold value. However, the limited works already reported in literature leave many questions open. There is still room for investigations and introduction of new technologies as no single technique can threshold all kinds of images. Also, the fusion of fuzziness and opposition is an intriguing field that has not been explored so far.

Since the idea of opposition in image processing is new. The results presented in [32] show some improvement in the outcome of the image thresholding task, and in view of the fact that looking at two segments as opposites seems to contain some potential for a non-conventional formulation of the pixel classification, we have decided to focus on OFT algorithm and aim for increased accuracy. The accuracy in this work is quantified as the agreement between the generated segment and the "gold standard image". The latter is a binary image prepared by the expert. Hence, accuracy in this thesis is directed toward expert-based evaluation.

The purpose of this thesis is to increase the accuracy of the OFT algorithm. We introduce a weighting factor that defines the value of the information in each local region of the image and guides the algorithm toward a superior threshold value which results in more accurate outcomes. This weight factor is calculated based on simple characteristics of the region, namely mean and standard deviation, which makes it easy to calculate. It also adds no complexity to the implementation and computational time.

1.3 Outline

The reminder of this thesis is organized as follows. In Chapter 2, we provide a background on image thresholding, the concept, different approaches and their advantages and challenges. Two traditional and widely used thresholding methods, Otsu and K-Means, are explicated, and then we review the oppositional fuzzy thresholding method (OFT) that is the foundation algorithm of this thesis. Chapter 3 is devoted to opposition based computing idea, its definitions and a summary of related research in this field. Our proposed method, weighted opposition fuzzy thresholding (WOFT), is discussed in Chapter 4. Experimental results demonstrating the performance of our approach are presented on synthetic and real world ultrasound images. Accuracy measurements are compared and discussed with the results driven from Otsu, K-Means, and OFT methods. As a final point, in Chapter 5, we

draw conclusions from our proposed method and discuss potential future directions for our work.

CHAPTER 2

Image Thresholding

Image segmentation plays a crucial rule in many computer vision applications. It refers to the technique of separating image pixels and forming different regions by identifying a similarity factor among their features. In most cases due to various factors, the object in the image is difficult to segment. The simplest, yet often most effective method of segmentation is "Thresholding" which is the classification of image pixels to black and white regions based on their intensity.

2.1 Background Review

Thresholding is extensively used as a preliminary step to separate the image background and foreground (object). The principal idea is to find an optimal threshold value that can separate the object and background pixels. In an image with L number of gray levels, using a threshold value of T, the thresholding task can be applied by the following rule:

$$g_i(T) = \begin{cases} g_O = 1 & for \quad 0 \le g_i \le T, \\ g_B = 0 & for \quad T < g_i \le L - 1. \end{cases} \tag{2.1}$$

g_O and g_B represent object and background gray levels, respectively. This idea will extend to Multilevel Image Thresholding [52] techniques in which more than one threshold value is set to segment the image into several classes.

Figure 2.1 illustrates an example of thresholding an ultrasound image with two different threshold values. In (c) all the pixels with gray level intensity smaller than 60 represent the "Object" while in (d) gray level 100 has been used as a threshold.

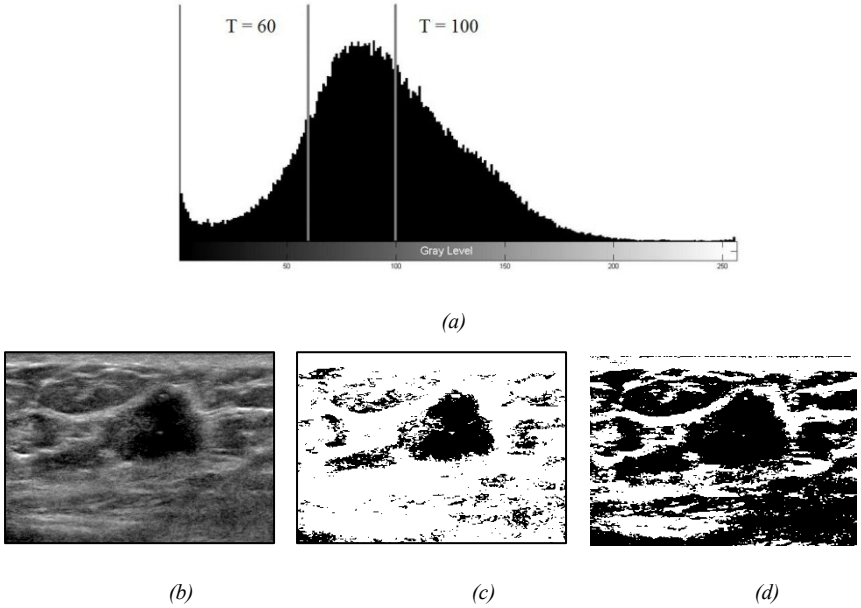


Figure 2.1: Image Thresholding - (a) represents the histogram of image (b). The thresholded images using two values of T=60 and T=100 are presented in (c) and (d), respectively.

Finding the optimal thresholding value is the main focus of different thresholding techniques. In every thresholding application, the shape of histogram, the image content, and the requirements of the application have a definite impact on this task [53, 10].

Since the emphasis of this thesis is not on thresholding techniques in general but enhancing of an oppositional fuzzy thresholding technique, only a brief review of studies conducted in the field of thresholding is presented followed by a more detailed explanation of two well known thresholding methods that are used as benchmark functions to compare the experimental results of our proposed method with. The "Quasi-global Oppositional Fuzzy Thresholding" method is then explained, as one of the thresholding techniques, which is the basis for this work.

Based on overviews conducted in [35] and [39], thresholding methods can be categorized into following groups:

- ***Histogram shape-based methods:***

  This group of methods is based on the shape properties of the image histogram. For example, Resenfeld's method [30] calculates the *convex hull* of the histogram to get the deepest concavity points as the threshold. Other methods dealt with histogram valley-seeking problem [31]. *Peak-and-Valley thresholding* is another branch of this group that focuses on twisting the histogram function using a smoothing and differencing kernel and searching for optimum threshold point in zero crossing of the histogram.

- ***Clustering and Classification-based methods:***

  The initiative of this group of algorithms is built on the *clustering analysis* of the gray level data. They are considered to be unsupervised techniques since prior knowledge is not part of the analysis. The main task of this type of algorithms is to create different image regions that any pixel of the image belongs to only one of them. Otsu's method [21] is a popular technique in this group that finds the optimal threshold by minimizing the weighted sum of within-class variances for background and foreground pixels. *Iterative thresholding* is another case in this group. Riddler's scheme [29] is the first identified model of two-class Gaussian mixture that establishes the threshold anchored in average of the foreground and background

class means. More recent practices have been used *Fuzzy clustering* ideas for thresholding. The works presented in [12], [28], and [32] used fuzzy memberships based on pixels' distance from each class's mean to define which class a pixel belongs to and subsequently define the threshold as the cross over point of membership functions. Furthermore, *Classification thresholding* methods [33], [41] try to classify image pixels based on their characteristics in such a way that pixels in one class have highest degree of similarity to each other while pixels that belong to different classes are very unlikely similar.

- ***Entropy-based methods:***

  This group of algorithms takes advantage of entropy of the distribution of image's gray levels. Several studies have been conducted on *Entropic thresholdin*g [13] that their main focus is to maximize the sum of entropy values of foreground and background regions of an image that lead them to the optimal threshold value. Others [15], [16] exploit *cross-entropic thresholding* techniques as the minimization of an information theoretic distance. *Fuzzy entropic thresholding* is another branch of this group. In different research activities, [36], [6] for example, fuzzy memberships are used to signify the strength of the relationship of a gray level to the foreground or background regions. The optimal threshold, then, is defined as the value that minimizes the sum of the fuzzy entropies of two regions.

- ***Object attribute-based methods:***

  Threshold values in these techniques are defined based on some attribute or similarity measures between the original image and its binarized version. Edges, shape compactness, connectivity, or texture are examples of attributes that are used in these types of methods [48], [22], and [20]. The fact that fuzzy methods have the ability to present subjective information has motivated some studies to benefit from fuzzy membership values establishing various fuzzy measures in fuzzy similarity thresholding procedures.

- ***Local, global and hybrid methods:***

  This group of methods is based on only locally [33] or globally selected operations or information in an image. Combination of local and global characteristics of the image is used in the hybrid methods.

- ***Spatial and feature-based methods:***

  In the group of spatial-based methods, any point in the image is characterized with its dependency within a neighborhood. These techniques assume that pixels in an object are spatially close. In most cases this assumption leads to better results in image segmentation / thresholding. The focus for feature-based algorithms is on employing an image feature to extract a homogeneous region in the feature space. [14] is considered as one of the pioneers in this group.

- ***Intelligent methods:***

  This group of algorithms has used different machine learning techniques for image segmentation / thresholding tasks. Artificial Neural Networks [9] and Reinforcement Learning [33] are used in different ways in image segmentation problems. Based on their nature, neural networks need some sample data for training purpose while reinforcement learning has no need for training samples [40].

2.2 Otsu Thresholding

One of the common image processing tasks is to transfer a grayscale image to its monochrome version. Otsu's method, named after its inventor Nobuyuki Otsu [21], is one of the most popular automatic binarization algorithms, which is used in many computer vision and medical imaging applications. Otsu's method is based on the shape of the histogram and is considered as a non-parameterized algorithm since it obtains the optimal threshold by maximizing the between-class variance with a comprehensive search. It also can be extended to multi level thresholding that will result in segmentation.

This thresholding method iterates trough all the possible threshold values. In each iteration, the weighted within-class variance of two classes (foreground and background on either sides of the threshold point) gets calculated. The optimal threshold is the one that minimizes the within-class variance, hence, maximizing the between class variance.

Consider an image with N pixels with gray levels in $[1, \dots, L]$ interval. To divide this image into two classes, C_1 with gray levels $[1, \dots, t]$ and C_2 with $[t+1, \dots, L]$, we need to calculate the gray level distribution for both classes as:

$$\begin{aligned} C_1 &: \quad p_1/\omega_1(t) \, , \dots , \, p_t/\omega_1(t) \\ C_2 &: \quad p_{t+1}/\omega_2(t) \, , \dots , \, p_L/\omega_2(t) \end{aligned} \tag{2.2}$$

where

$$\begin{aligned} \omega_1(t) &= \sum_{i=1}^{t} p_i \\ \omega_2(t) &= \sum_{i=t+1}^{L} p_i \end{aligned} \tag{2.3}$$

Here, p_i represents the probability of gray level i with f_i pixels and is calculated as:

$$p_i = f_i/N \tag{2.4}$$

Now μ_1 and μ_2 , the mean values of two classes, can be estimated as:

$$\begin{aligned} \mu_1 &= \sum_{i=1}^{t} i \, p_i/\omega_1(t) \\ \mu_2 &= \sum_{i=t+1}^{L} i \, p_i/\omega_2(t) \end{aligned} \tag{2.5}$$

Let μ_T be the mean intensity of the whole image. It is easy to show that:

$$\mu_T = \omega_1\mu_1 + \omega_2\mu_2 \tag{2.6}$$

and

$$\omega_1 + \omega_2 = 1 \tag{2.7}$$

Otsu defines the intra-class variance of the thresholded image as:

$$\sigma_B{}^2 = \omega_1(\mu_1 - \mu_T)^2 + \omega_2(\mu_2 - \mu_T)^2 \tag{2.8}$$

Otsu verified that for bi-level thresholding, the optimal threshold t_{opt} is chosen so that $\sigma_B{}^2$ is maximized; that is,

$$t_{opt} = Arg\ Max\{\sigma_B{}^2(t)\} \qquad 1 \le t < L \tag{2.9}$$

This method is easy to implement and since it operates on histograms, which are arrays of length 256, it considered being quiet fast. However, assumption of uniform illumination and bimodal histogram are the two drawbacks counted for this technique. Table 2.1 describes Otsu's algorithm.

Read image I with gray levels of $[1, \dots, L]$
Compute histogram and probabilities of each intensity level
Initialize $\omega_i(0) = 0$
For each $t = 1\ to\ L$
Calculate ω_i and μ_i
Update $\sigma_B{}^2(t)$
t_{opt} is the t that corresponds to the maximum $\sigma_B{}^2(t)$

Table 2.1: Pseudo-code of Otsu's algorithm

2.3 K-means Thresholding

K-means [17] is known as one of the most frequently used unsupervised learning algorithms to solve the clustering problems. It is a simple algorithm that classifies a dataset of observations to k clusters fixed *a priori*. In the case of $k = 2$, the algorithm can be used as a thresholding technique.

The process uses an iterative refinement technique. It starts with defining centroids for each cluster as much as possible far away from each other. These centroids may be specified randomly or by some heuristic. Then each observation point x_j gets associated to the nearest centroid until no point is pending. New centroids can then be calculated for the resulted clusters which can be counted as new cluster means to associate the observation points to. Through these iterations the centroids change their location until no movement is done anymore.

As a final step, the process tries to minimize an objective function, in this case within-cluster sum of squares:

$$Arg\ Min \sum_{i=1}^{k} \sum_{x_j \in C_i} \|x_j - \mu_i\|^2 \tag{2.10}$$

where μ_i represents the mean of points in cluster C_i.

The algorithm of this method is illustrated in Table 2.2. This algorithm is extensively sensitive to the initially selected cluster centers that may results to finding suboptimal clusters. Multiple runs will reduce this effect but will lead to longer computational time. The main disadvantage of K-means algorithm is that the results depend on the value of k and in most problems there is no way of knowing the number of clusters. This, however, is not a concern when using this process as a thresholding technique.

Read $I = \{i_1, \dots, i_p\}$ as a set to be divided into n clusters

Set $C = \{C_1, \dots, C_n\}$ (cluster centroids) to random selections of I

For $j = 1\ to\ p$

    For $k = 1\ to\ n$

        $m_{i_j} = argmin\ distance(i_j, C_k)$

    Repeat until m is not changed

        For $k = 1\ to\ n$

            Re-compute i_k as the centroid of $\{i \mid m(i) = k\}$

        For $j = 1\ to\ p$

            For $k = 1\ to\ n$

                $m_{i_j} = argmin\ distance(i_j, C_k)$

Return C

Table 2.2: Pseudo-code of K-means algorithm [2]

Some examples of images that are thresholded using Otsu and K-means methods are presented in second and third rows of Figure 2.2.

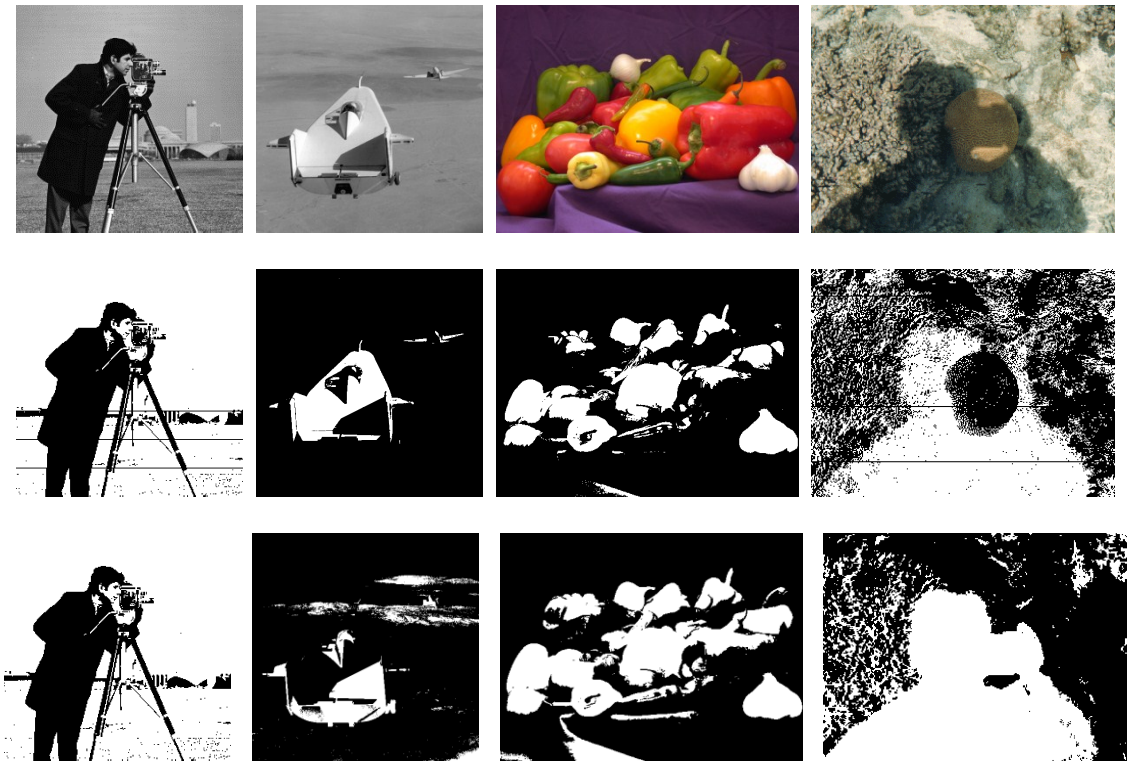


Figure 2.2: Image samples resulted from Otsu and K-means thresholding.

2.4 Oppositional Fuzzy Thresholding

Tizhoosh et al. introduced this method as a "proof of concept for usefulness of oppositeness in context of fuzzy set theory" [32]. The idea of opposite fuzzy sets was always engaged to fuzzy systems; however, the concept of oppositeness for a fuzzy set is not as apparent as expected and it may be mistaken by the perception of "negation". For example, in a gray scaled image, one may consider "very bright" and "not very bright" as opposites while the true opposite of "very bright" is "very dark". This algorithm exploits the relationship between fuzzy and its corresponding opposite fuzzy set of "dark" and "bright" to threshold a digital image. To better understand the logic of the algorithm, let's first clarify what an opposite fuzzy set is.

Definition (Fuzzy Set) – A fuzzy set $A \subset X$ is defined as:

$$A = \{(x, \mu_A(x)) \mid x \in X,\ \mu(x) \in [0,1]\} \tag{2.11}$$

The $\mu_A(x)$ represents the membership function and is given as:

$$\mu_A(x) = f(x;\ \boldsymbol{a}, \delta) \tag{2.12}$$

where $\mu_A(\boldsymbol{a}) = 1 \quad \forall\ a_i \in \boldsymbol{a}$ and δ is the parameter that changes the shape of the membership function. Basically, this is a function of three parameters with one of them, namely x, being the actual variable.

Definition (Opposite Fuzzy Set) – Opposite fuzzy set $\breve{A} \subset X$ for a given fuzzy set $A \subset X$ is defined as:

$$\breve{A} = \{(x, \mu_{\breve{A}}(x)) \mid x \in X,\ \mu_{\breve{A}}(x) \in [0,1]\} \tag{2.13}$$

where $\mu_{\breve{A}}(x) = f(x;\ \breve{\boldsymbol{a}}, \breve{\delta})$.

The vector $\boldsymbol{a} = [a_1, a_2, \dots]$ and its opposite vector $\breve{\boldsymbol{a}} = [\breve{a}_1, \breve{a}_2, \dots]$ represent the points in the universe of discourse with $\mu(a_i) = \mu(\breve{a}_i) = 1$.

Definition (Type I Opposite Fuzzy Set) – The set $\breve{A}_I$ with the membership function $\mu_{\breve{A}_I}(x) = f(x;\ \breve{a}, \breve{\delta})$ is considered to be the type I opposite of the set A if $\breve{a}$ and $\breve{\delta}$ are type I opposites of a and δ, respectively.

Definition (Type II Opposite Fuzzy Set) – The set $\breve{A}_{II}$ with the membership function $\mu_{\breve{A}_{II}}(x) = f(x;\ \breve{a}, \breve{\delta})$ is considered to be the type II opposite of the set A if $\breve{a}$ and $\breve{\delta}$ are type II opposites of a and δ, respectively.

Knowing the definitions of opposite fuzzy sets, now we can discuss the OFT method. To threshold an image, the process starts with capturing a sample point of user's choice, ideally within the object of interest. The algorithm starts to create co-centered windows around this pixel. A predefined parameter, d, is the gap between the neighboring windows that are constructed around the central point. In this experiment the value of 10

pixels has been empirically chosen for this parameter ($d = 10$). For each window, a fuzzy set and an opposite fuzzy set are calculated to characterize the dark and bright areas hence object and opposite object (background). Then the entropy difference between each fuzzy set and its opposite is calculated. When all the measurements for all the sub-images are performed, the window that delivers the maximum (or minimum) entropy difference is located and based on the membership functions of this window two representative numbers are produced which the optimum threshold value will be quantified as the average of them. The image then gets thresholded using calculated threshold value. A cleaning process is applied to thresholded image that labels 8-connected objects and sets the value of 1 to the components that are part of the region which central pixel was selected from. Then all the dark areas surrounded by light pixels (holes) are filled to create an object of homogeneous pixels.

The basic principle of OFT is illustrated in Figure 2.3. The pseudo-code of this method is presented in Table 2.3. Also, The pseudo-code of the cleaning step that is used in both OFT and WOFT algorithms is present in Table 2.4.

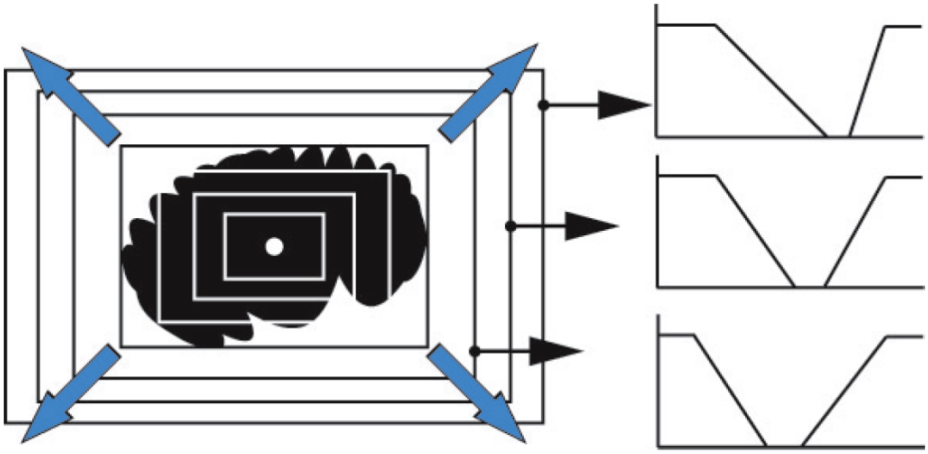


Figure 2.3: Basic principle of OFT: Starting from a point (the white circle), co-centeric windows are constructed around the object (black shape) in increasing order. For each window membership functions for object (dark) and anti-object (bright) are formed (only three last membership function pairs are shown here). The entropy (fuzziness) difference between object and anti-object are kept to find the minimum and maximum entropy differences. The corresponding membership values are then used to calculated thresholds. (Adopted from [32])

Read image I of size $R \times C$

Pre-process I if necessary

Initialize parameters

Acquire the coordinate of object's center (x_c, y_c)

For $\delta = \delta_{min} : d : \delta_{max}$

$x_1 = x_c - \delta$ and $y_1 = y_c - \delta$

$x_2 = x_c + \delta$ and $y_2 = y_c + \delta$

Copy $I(x_1 : x_2, y_1 : x_1)$ into I_{sub}

Calculate $h = histogram(I_{sub})$ and normalize it

$g_{min} = \min(I_{sub})$ and $g_{mid} = \text{mean}(I_{sub})$

Construct object O with a Z-MF and calculate its fuzziness

$E_O(i) = \frac{1}{RC} \sum h(g_i) \min(O, 1 - O)$

Construct anti-object $\breve{O}$ with a S-MF and calculate its fuzziness

$E_{\breve{O}}(i) = \frac{1}{RC} \sum h(g_i) \min(\breve{O}, 1 - \breve{O})$

Calculate the entropy difference $\eta(i) = |E_O(i) - E_{\breve{O}}(i)|$

Update $\eta_{max} \rightarrow O_{max} = O,\ \breve{O}_{max} = \breve{O},\ h_{max} = h$

Calculate the maximum threshold T_{max}

$$T_{max}^{O} = \frac{\sum h_{max}(i)\ O_{max}(i)}{\sum h_{max}(i)}$$

$$T_{max}^{\breve{O}} = \frac{\sum h_{max}(i)\ \breve{O}_{max}(i)}{\sum h_{max}(i)}$$

$$T_{max} = 255 \times \frac{T_{max}^{O} + T_{max}^{\breve{O}}}{2}$$

I_T = Thresholded image using T_{max}

Call Cleaning step for I_T with respect to (x_c, y_c) to extract the object

Table 2.3: Pseudo-code of OFT algorithm

Initialize L and $OBJECT$ as zero matrices of the same size as I_T
Set L to labels for the 8-connected objects in I_T
$L(x_c, y_c)$ defines the label number of the desired object
Find the elements of L that have the same label as $L(x_c, y_c)$
Mark those elements in $OBJECT$ with the value of 1
Fill the holes of the $OBJECT$

Table 2.4: Pseudo-code of Cleaning step used in both OFT and WOFT methods

It is noticeable that the cleaning step forces the algorithm and its outcome to be very dependent on the user defined central point. If this central point is selected from outside of the region of the interest (object), the resulted thresholded image would not represent the object even though it's before cleaning version has the object as part of the thresholded region.

CHAPTER 3

Opposition-Based Computing

The concept of Opposition-Based Learning (OBL), an approach to improve the performance of machine learning algorithms, has been introduced by Tizhoosh [42-44]. The idea behind this technique is to increase the coverage of solution space by considering "opposites" to achieve a higher accuracy or a faster convergence.

3.1 Theory

According to The American Heritage Dictionary, opposition is "The relation existing between two propositions having an identical subject and predicate but differing in quantity, quality, or both".

Understanding and formalizing the logic of the opposition has a strong root in human history. Many examples of opposition exist in the world around us. This includes the simple concepts such left/right, in/out, cold/warm and extends to complex actions as social revolutions.

In the system of Aristotelian logic, the logical relation of the propositions of a system has been represented as the Square of Opposition (Figure 3.1). This diagram is a

simple visualization of the different ways that each of the propositions of a system is logically related, or opposed, to others.

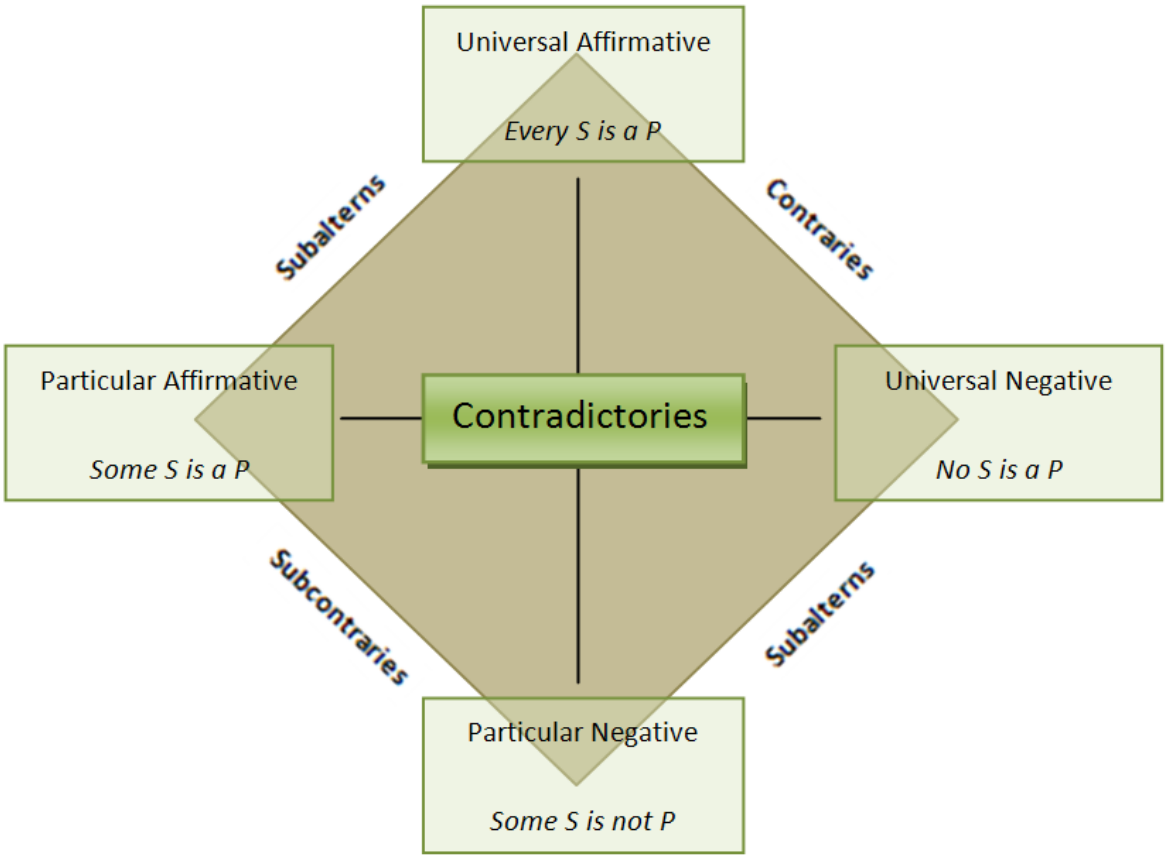


Figure 3.1: Square of Opposition

Even though opposition has been conversed vastly in the study of philosophy, this is not the only area of human studies that is touched with this concept. Natural Language [4], psychology [23], physics [3], and mathematics [19] are examples of different disciplines that have benefited from the concept of opposition in one way or another.

Approximation is a natural element of human life. In Science, usually $\hat{x}$ is an approximate solution to a particular estimation problem. This approximation could be

driven from knowledge or a random guess in case of some complex problems. In most cases we are aimed for higher precision, thus the estimated value must be modified to move closer to optimal value which forces us to deal with computational complexity. In any estimation algorithm, in the absence of *a priori* knowledge, all directions of the solution space should be covered. If the algorithm is searching for the solution of x, and one agrees that searching in the opposite direction could be beneficial, then we need to know how to calculate an opposite number $\breve{x}$.

Definition (Opposite Number) – Let $x \in R$ be a real number defined on a certain interval $x \in [a, b]$. The opposite number $\breve{x}$ is defined as follows:

$$\breve{x} = a + b - x \tag{3.1}$$

Analogously, the opposite number in a multidimensional case can be defined.

Definition (Type-I Opposite Points) – Let $P = (a_1, a_2, \dots, a_n)$ be a point in an n-dimensional coordinate system where $a_i \in \left[X^i_{min}, X^i_{max}\right] \in R$ and $\forall i \in [1, n]$. The Type-I opposite point $\breve{P}$ is completely defined by its coordinates $(\breve{a}_1, \breve{a}_2, \dots, \breve{a}_n)$ where:

$$\breve{a}_i = X^i_{min} + X^i_{max} - a_i \tag{3.2}$$

Following special cases exist when $n = 1$:

$$\begin{aligned} \breve{a} &= -a \quad for \quad X_{max} = -X_{min} \\ \breve{a} &= 1 - a \quad for \quad X_{max} = 1 \ and \ X_{min} = 0 \\ \breve{a} &= a \quad for \quad \frac{X_{max} + X_{min}}{2} = a \end{aligned} \tag{3.3}$$

Type-I opposition is a simple and basic understanding of this concept that can be beneficial to linear or symmetric systems. Dealing with complex and nonlinear relationships leads us to a higher level of opposition.

Definition (Type-II Opposite Points) – Let $y = f(x_1, x_2, \dots, x_n) \in R$ be an arbitrary function where $y \in [y_{min}, y_{max}]$. The type-II opposite point $\breve{P} = (\breve{a}_1, \breve{a}_2, \dots, \breve{a}_n)$ associated to every point $P = (a_1, a_2, \dots, a_n)$ is defined as:

$$\breve{a}_i = \{ x \mid \breve{y} = y_{min} + y_{max} - y\} \tag{3.4}$$

It is assumed that for the unknown function $f(x_1, x_2, \dots, x_n)$, y_{min} and y_{max} are given or can be estimated. On the other hand, according to

$$\breve{a}_i(t) = \left\{ x \mid \breve{y}(t) = \min_{j=1,\dots,t} y(j) + \max_{j=1,\dots,t} y(j) - y(t) \right\} \tag{3.5}$$

and if only the evaluation function $g(X)$ is available, then a temporal *degree of opposition* $\breve{\Phi}$ for any two variables x_1 and x_2 can be defined as:

$$\breve{\Phi}(x_1, x_2, t) = \frac{|g(x_1) - g(x_2)|}{\max\limits_{j=1,\dots,t} g(x_j) - \min\limits_{j=1,\dots,t} g(x_j)} \in [0, 1] \tag{3.6}$$

The type-II opposite $\breve{x}_i$ of any variable x_i can be given as:

$$\breve{x}_i = x_j \mid \breve{\Phi}(x_i, x_j, t) = \max_k \breve{\Phi}(x_i, x_k, t). \tag{3.7}$$

Definition (Degree of Oppositeness for Type-I Opposition) – The degree of oppositeness for two real numbers $a, b \in X$ bounded in $[X_{min}, X_{max}]$ can be calculated as:

$$\breve{\varphi} I(a, b) = \left(\frac{|a - b|}{X_{min} - X_{max}} \right)^{\beta} \tag{3.8}$$

where opposition intensity gets controlled by $\beta \in (0,1]$.

Definition (Degree of Oppositeness for Type-II Opposition) – Given evaluation function $g(\cdot)$, the degree oppositeness for two real numbers $a, b \in X$can be formulized as:

$$\breve{\varphi} II(a, b) = \left(\frac{|g(a) - g(b)|}{g_{max} - g_{min}} \right)^{\beta} \tag{3.9}$$

where opposition intensity gets controlled by $\beta \in (0,1]$.

If the bounds of the function $g(\cdot)$ are unknown, a temporal type-II opposition can be calculated:

$$\breve{\varphi}_{II}^{(t)}(a,b) = \left(\frac{|g(a) - g(b)|}{g_{max}^{(t)} - g_{min}^{(t)}}\right)^{\beta} \tag{3.10}$$

Having introduced the fundamental definitions, now we can define the opposition-based computing.

Definition (Opposition-Based Computing) – Let $f(x_1, x_2, \dots, x_n)$ be the unknown function in focus and $g(\cdot)$ a proper fitness evaluation function with higher values being desirable. If vector $X = (x_1, x_2, \dots, x_n)$ is an initial guess and $\breve{X} = (\breve{x}_1, \breve{x}_2, \dots, \breve{x}_n)$ its opposite value, we calculate $f(x)$ and $f(\breve{x})$ in every iteration. The search continues with x if $g\big(f(x)\big) \geq g\big(f(\breve{x})\big)$, otherwise we proceed with $\breve{x}$.

Figure (3.2) demonstrates a visualization of applying OBC on a simple one-dimensional case. Here, $\breve{x}$, the opposite value of our initial guess x, leads us to more accurate result in the solution area. Of course, in this specific case, the simple definition of the opposite happens to be the right choice. However, for any unknown and nonlinear function, the proper definition of the opposite, or its iterative refinement, would be crucial.

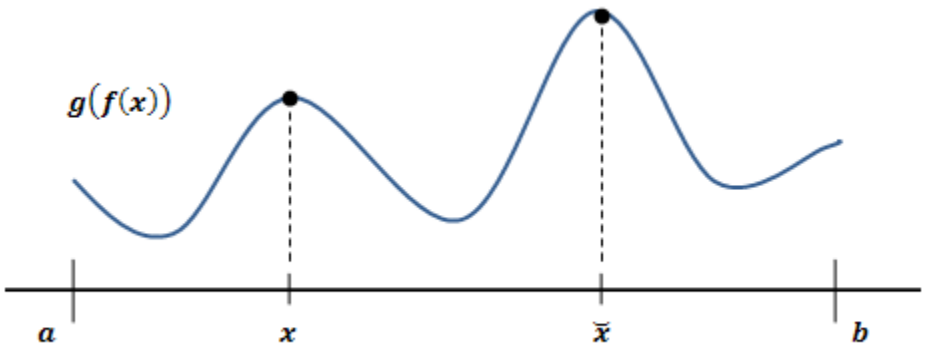


Figure 3.2: One dimensional illustration of OBC

The difference between type-I and type-II opposition is illustrated in Figure (3.3). In this case the goal is to find the extremes of a non-linear function $f : x \rightarrow [y_{min}, y_{max}]$. In this example for the initial guess x, comparing driven results from $\breve{x}_I$ and $\breve{x}_{II}$, the type-II opposite $\breve{x}_{II}$ corresponds to a closer result to the ultimate goal.

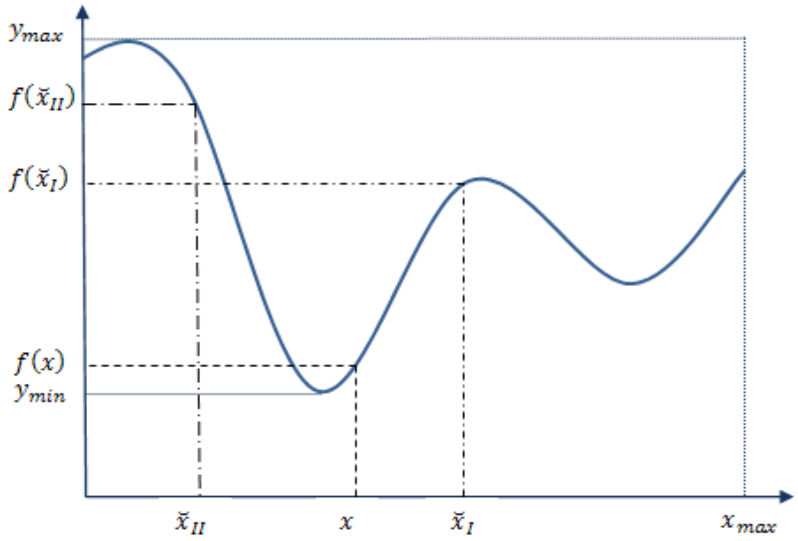


Figure 3.3: Type II OBC

3.2 Existing Opposition-Based Applications

The opposition idea has been successfully applied to optimization, learning, and image processing algorithms [47].

In the field of optimization, opposition is used to increase accuracy and speed in Ant Colony Optimization (ACO) [18]. In this technique, Opposite Pheromone per Node (OPN) and Opposite Pheromone Update (OPU) concepts have been introduced to increase accuracy in construction phase and speed in update phase respectively.

Differential Evolution (DE), which is a type of genetic algorithm, is another area that has been benefited from opposition [24], [25], [26], and [27]. In this approach by

embedding opposition-based population initialization and opposition-based generation jumping to the algorithm, the convergence speed of the algorithm has been accelerated.

In Reinforcement Learning (RL), which is a goal-directed method for solving problems in uncertain and dynamic environments, opposition has been used to accelerate the learning process [42], [37], and [38]. In this practice, the concept of Opposite Action [45] has been used that expedites the learning process for Q-Learning and Q(λ) methods.

In Neural Networks, Opposite Transfer Functions yield to establishing the Opposite Networks [49], [50], and [51]. The results of conducted works show significant improvement over standard backpropagation learning.

Image processing is another area that has been touched by opposition concepts. Works conducted in [46] and [32] perform using notion of Opposite Fuzzy Sets in Image Thresholding. The results from these works show satisfactory level of accuracy.

CHAPTER 4
Weighted Opposition-Based Fuzzy Thresholding

4.1 Proposed Method

As previously mentioned, the purpose of this work is to improve the oppositional thresholding via the increase in segmentation accuracy whereas the accuracy is equivalent to the similarity (or the degree of agreement) between the result image and its associated gold standard image. The oppositional fuzzy thresholding method (OFT) introduced by Tizhoosh et al. [32] was selected as the parent algorithm. This method tied up the concept of opposition-based computing to fuzzy sets to introduce the idea of "opposite fuzzy sets". Then it used the entropy difference between applied fuzzy sets to an image and their opposites to discover the optimal threshold value.

This method was considered as a starting point; however a higher level of accuracy was desired. For this purpose we introduced a local weight factor that is calculated for each sub-image in the process and quantifies the significance/validity of the calculated threshold value for each sub-image.

If the image is divided into N sub-images, the weight factor for each sub image IMG_{Sub_i} is calculated as follows:

$$W_i = \frac{N-(i-1)}{N(N+1)/2} \times \frac{Mean(IMG_{Sub_i})}{K \times Std(IMG_{Sub_i})} \quad 4.1$$

In every iteration of the process, this weight factor is calculated and multiplied with the fuzzy entropy of the object (e.g. foreground) and anti-object (e.g. background) to define the ultimate threshold value of the image. Based on this idea, the fuzzy entropy of the object and anti-object can be defined, respectively, as:

$$E_O(i) = \frac{1}{RC} \sum h(g_i)\, MIN(O_i, 1 - O_i) \times W_i \quad 4.2$$

and

$$E_{\breve{O}}(i) = \frac{1}{RC} \sum h(g_i)\, MIN\left(\breve{O}_i, 1 - \breve{O}_i\right) \times W_i \quad 4.3$$

The rationale behind the weight factor can be discussed based on the closeness of the sub-image to the user defined central pixel along with contrast of the pixels in the neighborhood of the sub-image.

As can be seen from equation 4.1, the calculated weight is based on two factors. The first factor expresses the closeness of the sub image to the centroid point. We assume that since the centroid point is selected by user within the object, the constructed sub-images around it have more characteristics of the object when they are closer to the central point. By incrementing the sub-image area through iterations, the-sub image would represent more characteristics of the whole image rather than of the object of interest. Thus the closer the sub-image to the central point the stronger its contribution should be to define the optimal threshold value.

On the other hand, the second element of the formulated weight relies on the fact that the local mean and standard deviation adopt the value of the threshold according to the contrast in the local neighborhood of the pixel.

The points of departure for introducing our weight factor are the following thoughts and assumptions: we know that the object of interest is a dark (or bright) area and logically the central point is selected from the pixels that are inside this area, ideally in the center of the object. As mentioned above, by creating co-centered windows around the central point, the process tries to define a threshold value that minimizes (or maximizes) the absolute value of the entropies of the object and opposite object. In this procedure, the windows at the beginning of the process are inside the object which illustrate a low contrast dark (or bright) region. When the window is over such a region, the mean will be quiet low (or high) and the standard deviation will be rather low. With these windows, despite the fact that closeness to the central point forces a high weight, the low/high mean and low standard deviation would reduce the weight factor to prevent the algorithm from defining the ultimate threshold based on the primary windows' characteristics. When the process starts to examine the windows that consist of both object and background regions, even though these windows are farther from the central point, but the higher mean of the sub-image would drive the weight to be considered strong enough to be able to contribute to the selection of the threshold. Furthermore, when calculating the weight for final windows in the image, the distance from central point has a negative effect on the weight minimizing it to force the process not to rely on the derived threshold based on their characteristics. Fundamentally, this formula tries to create a balance between closeness to central point and consideration of uneven sub-images to assist the algorithm with the estimation of the threshold value.

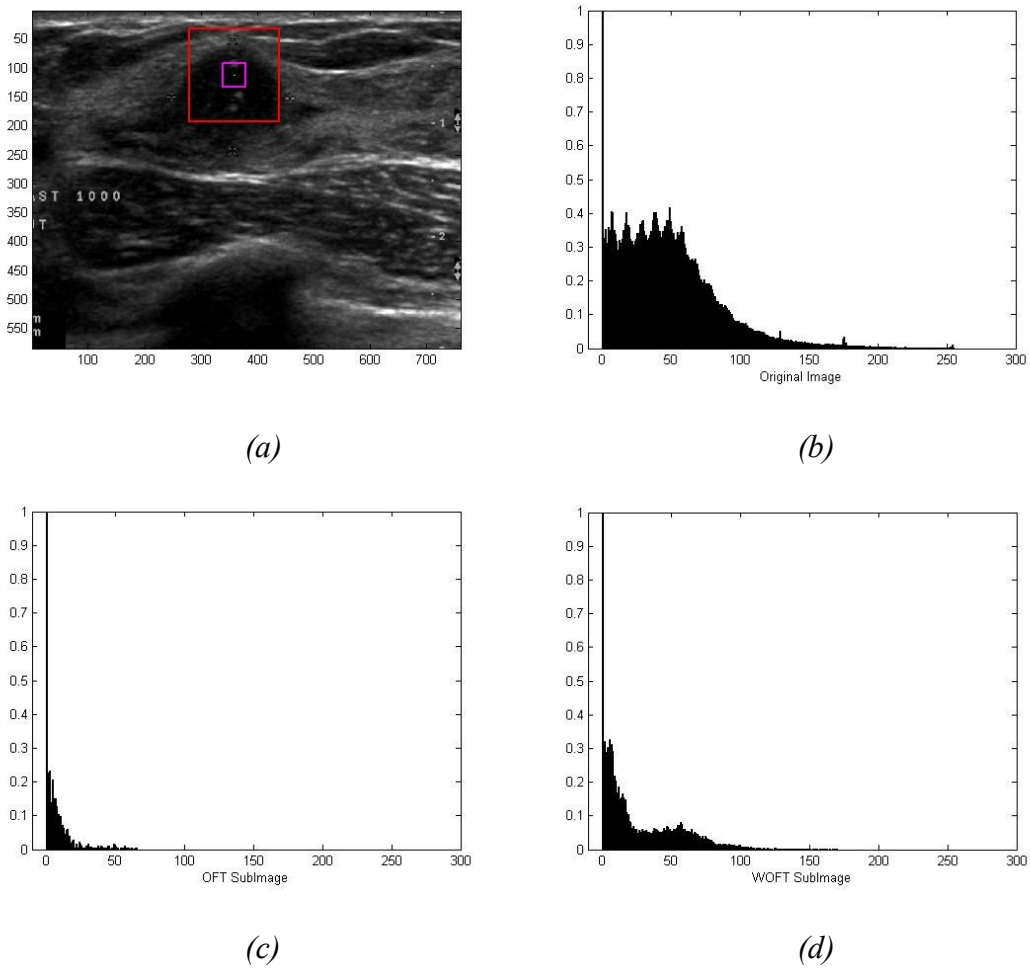
Figures 4.1 and 4.2 illustrate the difference between selections of the sub-image that lead to different threshold values from OFT and WOFT methods.

In Figure 4.1 (a) the central point with the coordinates of (359, 12) and two co-centered windows with coordinates of ((339, 92), (379, 132)) and ((279, 32), (439, 192))

are presented that the first window (in purple) is the target sub-image from OFT and the second (in red) from WOFT method. In this trial, the threshold value of 14 is calculated using OFT while WOFT resulted in thresholding the image with a value of 38. In WOFT, our presented weight factor pushed the process to ignore the value from the smaller window. The fact that this window has a small mean and standard deviation forced the method to continue until reaching a reasonable balance between closeness and being uneven in gray level histogram.

As it can be seen, the sub-image selected by our method has part of the object and some areas of the background which led to a more accurate thresholding.

The images resulted by the trial presented in Figure 4.1 are illustrated in Figure 4.2. In this figure the difference between two thresholding processes with respect to their resulted image, before and after cleaning step, is represented. Again, beside the accuracy measurements presented in the experimental results, the visual inspection also confirms that WOFT was more accurate in thresholding the object by relying on the introduced weight factor.



(a) *(b)*

(c) *(d)*

Figure 4.1: Comparison of OFT and WOFT methods with respect to selecting the sub-image that define the threshold value. (a) represents the original image and selected sub images by OFT and WOFT methods in purple and red respectively, The histograms of original image, OFT sub image, WOFT sub image are shown in (b), (c), and (d).

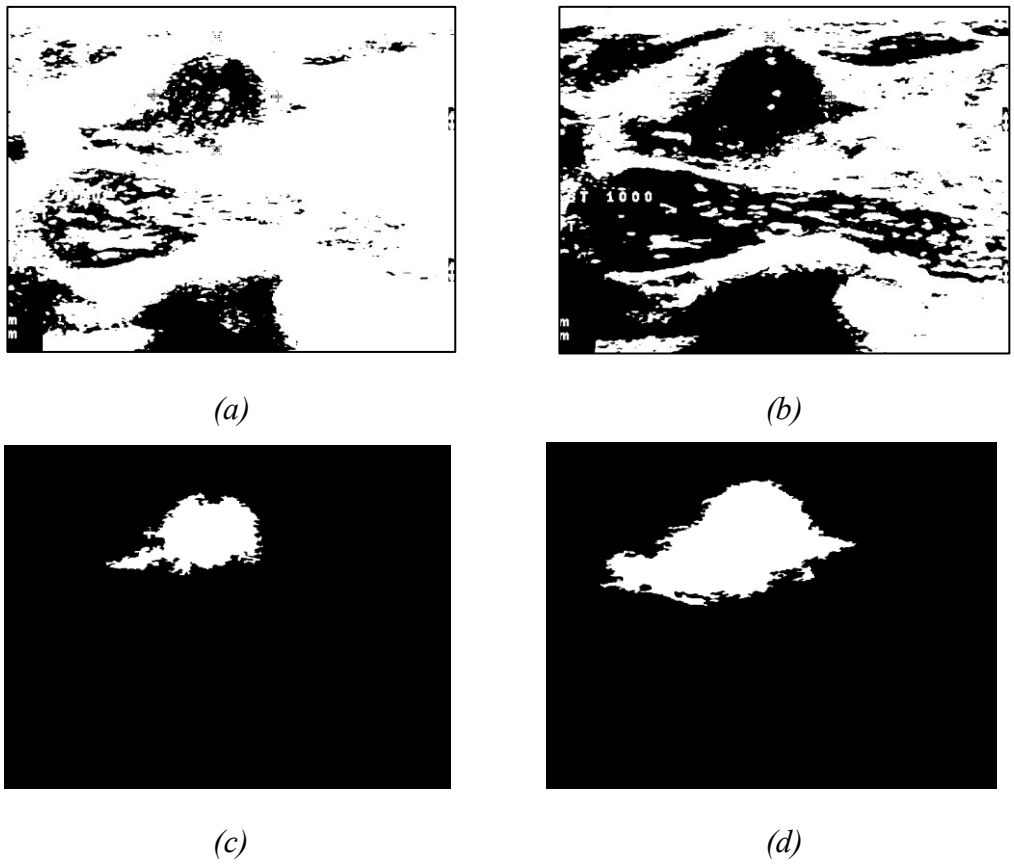


Figure 4.2: Comparison of OFT and WOFT methods with respect to selecting the sub image that defines the threshold value. (a) and (c) represent before and after cleaning resulted images form OFT, (b) and (d) show the same for WOFT.

Beside the weight factor, we also applied some image filtering techniques to prepare the image for thresholding task. Noisy images are a common dilemma in image processing applications. Since image thresholding or segmentation tasks are influenced by image noise, we decided to preprocess our test images with filtering and noise reduction techniques. Our test images are filtered with median filter and their contrast level has been increased by transforming the values using contrast-limited adaptive histogram equalization (CLAHE) [54].

4.2 Experimental Trials

After a discussion on the configuration and process of the proposed technique, we now dedicate this section to evaluate its performance. For this purpose, we have considered a variety of measurements on 20 synthetic images as well as 15 sample breast ultrasound images [1]. Synthetic images were used to compensate for lack of large number of real images. The breast ultrasound sample set consists of several ultrasound scans that contain anechoic (dark) breast cyst in conjunction with scans that contain non-cystic breast masses that should be examined for malignancy. The task of segmentation is relatively easy for the first group comparing to the second group that are identified as challenging to segment.

As mentioned before, the purpose of this thesis is not to propose a complete segmentation solution for breast ultrasound images. This category of images is chosen to be used as test cases since they are known to be difficult to segment and can perform an assessment to show the reliability of the proposed method in the real world applications. Synthetic images, on the other hand, are not as complicated as the other group yet problematical enough to be used as a challenge against the proposed method.

A manually segmented ground-truth, or gold standard image, have been provided by an image-processing expert for the sample images that has been used as the ultimate result of thresholding task in the conducted comparisons.

4.3 Experimental Setup

To challenge the robustness of the proposed algorithm, we threshold each sample image with 5 different central points. These sample points correspond to 5 pixels inside the area of interest. Test case 1 represents the experiment that the sample points are selected from the center of the object in all the test images. Test cases 2, 3, 4, and 5, respectively, correspond to trials with test points sampled from left, right, top and bottom regions of the

object. This will direct us to examine each image in 5 different ways which leads to 75 trials with ultrasound and 100 experiments with synthetic images.

Since this algorithm is not dependant on any parameters[1], no predefined setup is required, which can be considered as an advantage for this method.

Every performed trial results in two segmented images based on threshold values driven from maximum and minimum entropy differences for each image. The resulted images from the proposed method are compared to the results of the original algorithm to illustrate the effectiveness of the proposed modification to the original algorithm. Also, to demonstrate the robustness of the results, a comparison with two popular thresholding/segmentation techniques, namely Otsu and K-Means methods, is conducted.

In order to evaluate the performance, two measurements of similarity and diversity between the result image and its associated gold standard image are considered. The Jaccard Index [11], also known as Jaccard Similarity Coefficient, is the first measure. It is known as one of the most useful and widely used similarity indices for binary data. This method calculates the area of overlap, A_i , between the thresholded binary image, B_i , and its ground-truth image, G_i , as follow:

$$A_i = \frac{|B_i \cap G_i|}{|B_i \cup G_i|} \times \%100 \qquad 4.4$$

This measure is zero if the two images are disjoint; here it means they have no common pixels in segmented object, and is 1 if they are identical. Higher numbers means better conformity in the images.

The second method, Dice Coefficient [5], is another similarity measurement related to Jaccard Index. This method is named after Lee Raymond Dice and is calculated using the following equation:

[1] The original algorithm may vary by selecting different membership functions and entropy measures. However, the extension proposed in this work does not rely on any parameter.

$$D_i = \frac{2|B_i \cap G_i|}{|B_i| + |G_i|} \times \%100 \quad 4.5$$

Again with this measure, a value of zero indicates no overlap while 1 shows a perfect match between the resulted image and its gold standard. As it is apparent, this evaluation is similar to Jaccard but gives twice the weight to agreements. Furthermore, by applying both of these measures to resulted images, the goal is to get as close to 1 as possible.

To illustrate the results, each row on the tables presented in the following sections corresponds to the average and standard deviation of each test case for both methods. These factors stand for the resulting set of aforementioned measures for either 15 ultrasound or 20 synthetic images for each test case. Moreover, using t-test, for each test case the confidence interval and significance level is calculated. These estimates are advantageous since estimate of the mean varies from sample to sample; hence, instead of a single estimate of the mean the upper and lower limit for the mean is generated. This piece of information for each test case gives us an indication as how much uncertainty is involved with the estimate of accuracy with each method. Of course, a narrow and high interval demonstrates a more precise estimate. For this practice the confidence coefficient of 95% is used.

4.4 Results

In this section, a mixture of graphs and tables has been used to demonstrate the comparison of original and weighted oppositional fuzzy thresholding algorithms, OFT and WOFT. In order to be able to validate the efficiency of our algorithm and compare it with the original version of it, we performed a comparative experiment on our image set with regards to the different 5 cases of sample points for each image. To observe the effect of considering maximum or minimum entropy as the factors of thresholding process, each test case has been processed with both factors and the results are captured respectively.

4.4.1 Synthetic Images

In this section the result of applying OFT and WOFT algorithms on synthetic images are illustrated. These images have been created based on pre-defined binary images which have been corrupted by noise and filtered successively. Since synthetic images have lower level of complexity with respect to image thresholding or segmentation compare to ultrasound images, they show higher accuracy and similarity results. Tables 4-1 to 4-4 represent calculated average accuracy base on minimum and maximum entropy and average similarity, respectively. For each test case the confidence interval and significance level is calculated to better illustrate accuracy changes.

Table 4-1 - Area of Overlap based on Min Entropy Difference

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	79.41	6.81	[75.56 81.50]	1.83E-22	90.31	5.07	[87.95 94.12]	8.94E-23
2	75.64	7.11	[69.32 79.12]	1.91E-22	84.09	6.35	[80.15 86.72]	1.65E-22
3	78.69	6.51	[74.41 80.67]	1.79E-22	89.37	6.18	[86.41 91.83]	1.62E-22
4	82.19	6.92	[78.23 84.76]	1.89E-22	92.01	6.84	[88.59 95.02]	1.71E-22
5	79.81	7.64	[74.81 82.19]	1.97E-22	91.89	7.60	[88.07 93.68]	1.93E-22

Table 4-2 - Area of Overlap based on Max Entropy Difference

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	75.65	7.23	[69.41 81.18]	1.92E-22	85.32	6.84	[81.08 88.56]	1.82E-22
2	74.13	7.89	[67.15 80.57]	2.17E-22	80.94	7.5	[77.34 84.61]	2.13E-22
3	77.38	6.69	[74.28 81.40]	1.79E-22	88.67	6.13	[85.23 90.81]	1.73E-22
4	79.52	5.98	[75.76 83.11]	1.72E-22	90.11	5.93	[88.37 93.47]	1.70E-22
5	72.61	7.12	[68.17 76.53]	1.85E-22	85.47	6.73	[82.13 88.05]	1.77E-22

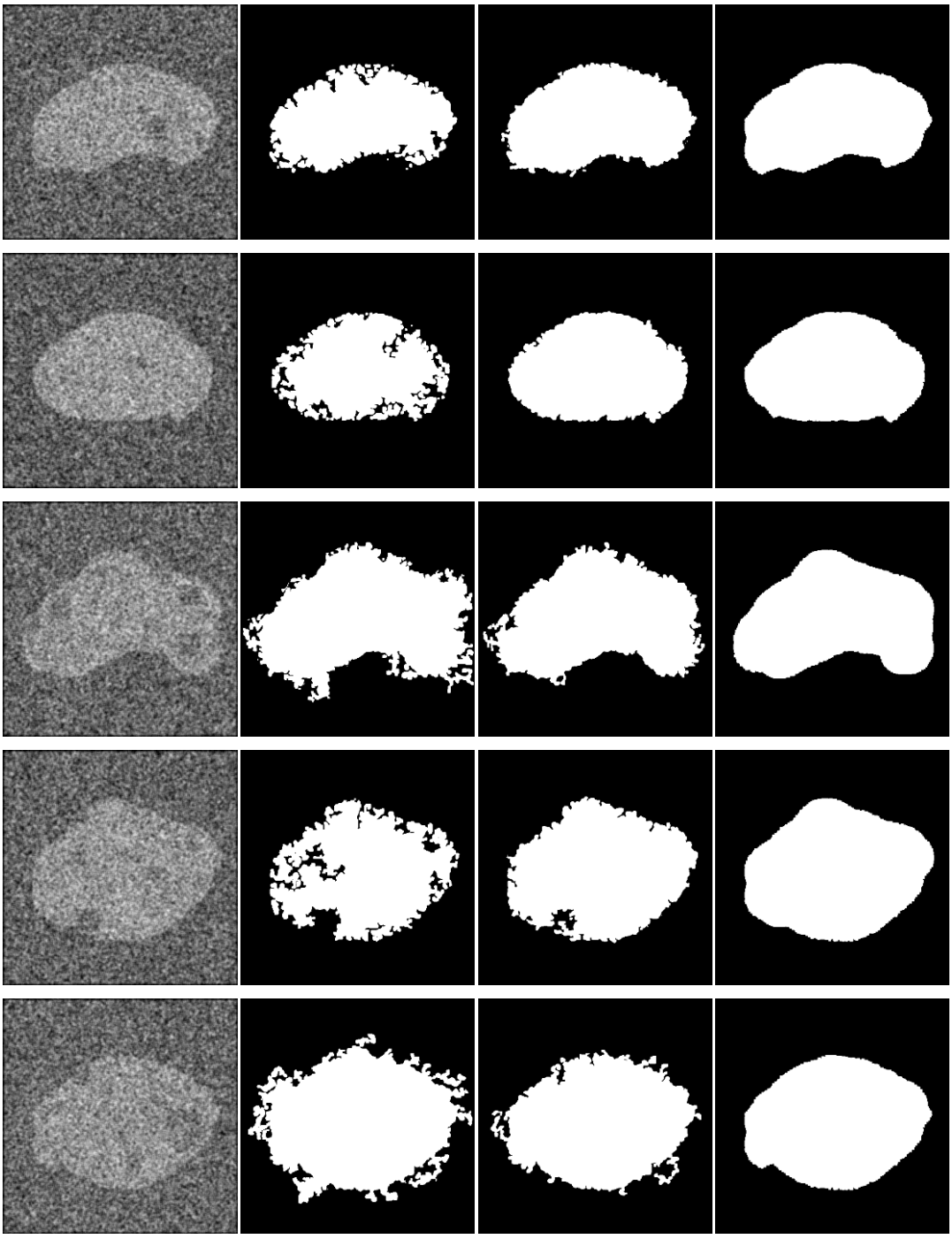
Table 4-3 - Similarity based on Min Entropy

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	81.33	7.19	[76.52 84.16]	1.83E-22	92.02	6.59	[87.69 95.28]	1.70E-22
2	77.43	8.49	[70.46 79.38]	2.37E-22	85.94	7.98	[81.39 87.41]	1.95E-22
3	80.12	8.29	[73.81 85.63]	2.19E-22	90.53	7.91	[86.18 93.08]	1.90E-22
4	83.44	7.86	[79.35 86.49]	2.05E-22	94.76	7.14	[89.37 96.45	1.88E-22
5	82.67	7.64	[78.64 84.19]	1.93E-22	92.82	6.89	[87.91 95.62]	1.76E-22

Table 4-4 - Similarity based on Max Entropy

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	78.30	7.49	[75.29 84.52]	1.83E-22	89.11	6.67	[86.56 94.15]	1.73E-22
2	75.94	7.81	[72.11 80.67]	1.91E-22	87.16	7.13	[84.63 91.08]	1.85E-22
3	80.12	8.03	[75.09 84.39]	1.97E-22	91.53	7.92	[88.19 95.36]	1.93E-22
4	83.44	7.64	[79.52 88.16]	1.89E-22	93.76	7.10	[89.82 97.27]	1.81E-22
5	82.67	7.37	[76.43 86.71]	1.97E-22	90.82	6.91	[87.34 93.38]	1.78E-22

The information presented in these tables confirms that the proposed method has higher rate of accuracy and similarity while even improving the confidence interval by making them narrower. The images presented in Figure 4.3 help us to visualize the increase in accuracy. The thresholded images, resulted from WOFT method, are shown in column c of the figure. Comparing these images to the resulted images from OFT in column b shows that they are closer to the gold standard images in column d.



(a) *(b)* *(c)* *(d)*

Figure 4.3: Thresholded synthetic images from test cases and their associated gold image are presented in columns (a) and (d). The resulted thresholded images from OFT and WOFT methods are presented in (b) and (c) respectively.

4.4.2 Breast Ultrasound Images

We used 15 breast ultrasound images to demonstrate the results of the proposed method. These images, as one of the most challenging cases in medical image analysis, are only used as a case study for this work. This thesis, hence, does not claim to provide a complete segmentation solution for breast ultrasound segmentation.

Tables 4-5 and 4-6 illustrate the average and standard deviation of Area of Overlap based on minimum/maximum entropy for each ultrasound image test case. These tables summarize the measurements that are captured after cleaning stage has been applied to the test images.

Table 4-5 - Area of Overlap based on Min Entropy Difference

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	65.96	31.7	[44.66 87.25]	4.19E-05	77.56	19.47	[64.48 90.65]	1.18E-07
2	67.35	29.13	[47.78 86.92]	1.70E-05	79.88	13.83	[70.60 89.17]	3.26E-09
3	65.26	32.79	[43.23 87.30]	6.07E-05	78.08	19.29	[65.12 91.04]	1.01E-07
4	64.29	31.98	[42.81 85.78]	5.59E-05	77.31	19.35	[64.31 90.31]	1.14E-07
5	69.93	27.05	[51.76 88.10]	6.38E-06	77.05	24.74	[60.42 93.67]	1.18E-06

Table 4-6 - Area of Overlap based on Max Entropy Difference

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	73.41	22.42	[58.35 88.48]	7.43E-07	77.56	18.23	[65.32 89.81]	6.27E-08
2	70.51	17.34	[58.86 82.16]	9.68E-08	78.56	18.67	[66.02 91.10]	6.98E-08
3	67.82	20.62	[53.97 81.68]	7.11E-07	76.93	19.78	[63.64 90.22]	1.48E-07
4	67.71	29.13	[48.14 87.28]	1.63E-05	72.06	26.38	[54.34 89.78]	3.90E-06
5	73.18	23.76	[57.21 89.14]	1.31E-06	76.02	16.85	[64.71 87.34]	3.57E-08

As can be seen, our WOFT method has increased the accuracy for the majority of the cases. Also, it is apparent that using minimum entropy as the base of calculation the threshold in the process leads us to higher accuracy rates.

Tables 4-7 to 4-8 follow the same arrangement but they represent the measurement of similarity (Dice Coefficient) between the resulted images and their gold standard image based on several sample points in each test case. The overall values support the judgment of the results illustrated in earlier tables.

Table 4-7 - Similarity based on Min Entropy

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	71.57	32.91	[49.46 93.68]	2.88E-05	85.84	15.43	[75.47 96.21]	4.72E-09
2	76.14	27.82	[57.45 94.83]	3.83E-06	85.62	15.26	[75.37 95.87]	4.33E-09
3	73.42	25.02	[56.61 90.23]	2.04E-06	83.54	15.6	[73.06 94.02]	6.82E-09
4	71.54	32.34	[49.82 93.27]	2.49E-05	85.78	15.4	[75.44 96.13]	4.65E-09
5	78.67	25.53	[61.52 95.82]	1.30E-06	83.38	21.94	[68.64 98.12]	1.84E-07

Table 4-8 - Similarity based on Max Entropy

Test Case	***OFT***				***EOFT***			
	Average	***STD***	***Confidence Interval***	***Significance Level***	***Average***	***STD***	***Confidence Interval***	***Significance Level***
1	81.61	14.91	[71.59 91.62]	5.52E-09	84.75	13.48	[75.69 93.80]	1.43E-09
2	79.66	18.7	[67.10 92.23]	6.20E-08	84.99	13.57	[75.87 94.11]	1.49E-09
3	78.21	20.17	[64.66 91.76]	1.52E-07	84.76	14.84	[74.80 94.73]	3.64E-09
4	79.22	20.02	[65.77 92.67]	1.26E-07	83.36	12.91	[74.69 92.04]	1.10E-09
5	77.01	25.27	[60.03 93.99]	1.44E-06	80.92	22.68	[65.68 96.16]	3.34E-07

Figure 4.4 presents some examples of thresholded images using OFT and WOFT algorithms for visual comparison. Additional examples are presented in Appendix A.

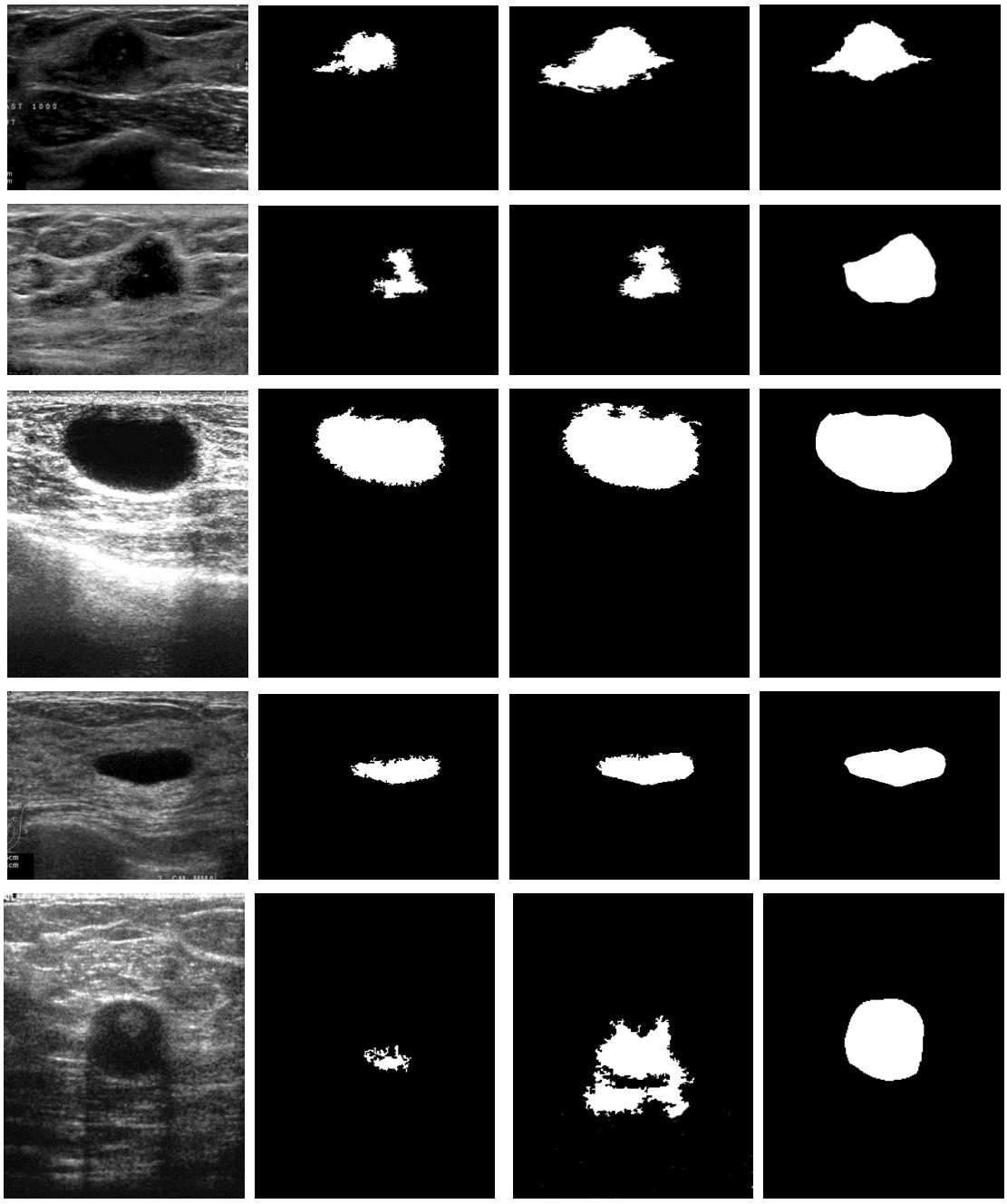


Figure 4.4: Thresholded ultrasound images. (a) and (d) are the original and Gold images. (b) and (c) demonstrate results of thresholding based on Minimum Entropy Difference with OFT and WOFT respectively.

Figure 4.5 provides a more detailed view of the comparison between accuracy of OFT and WOFT methods. This graph represents the Area of Overlap measures for the images in test case 1 trial which is calculated based on minimum entropy difference with cleaning process applied. As it is noticeable, in 13 cases out of 15, EOFT resulted in accuracy higher than or equal to OFT (12 of which have higher accuracy). As well, WOFT result set has smaller standard deviation and more desirable confidence intervals (Table 4.5), in comparison with OFT, which clearly demonstrates that the proposed method is more reliable than its parent algorithm.

The differences of the resulted images before and after cleaning stage for both OFT and WOFT methods have been illustrated in Figure 4.6. This figure is an easy visualization of how the cleaning phase, based on the area around the central pixel, increases the accuracy of the final result of the algorithm.

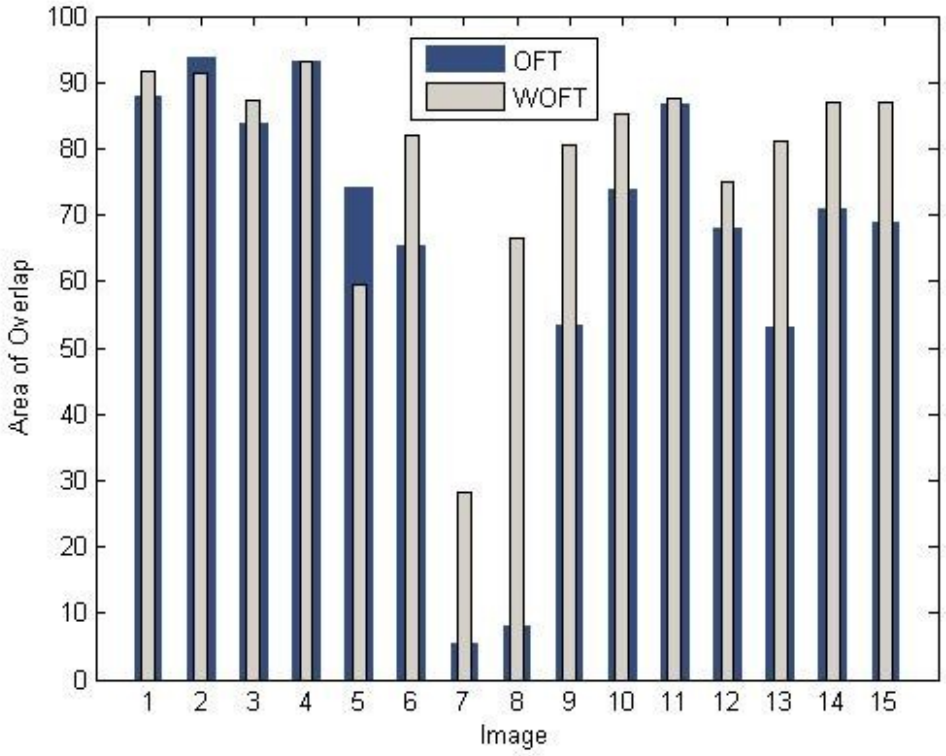
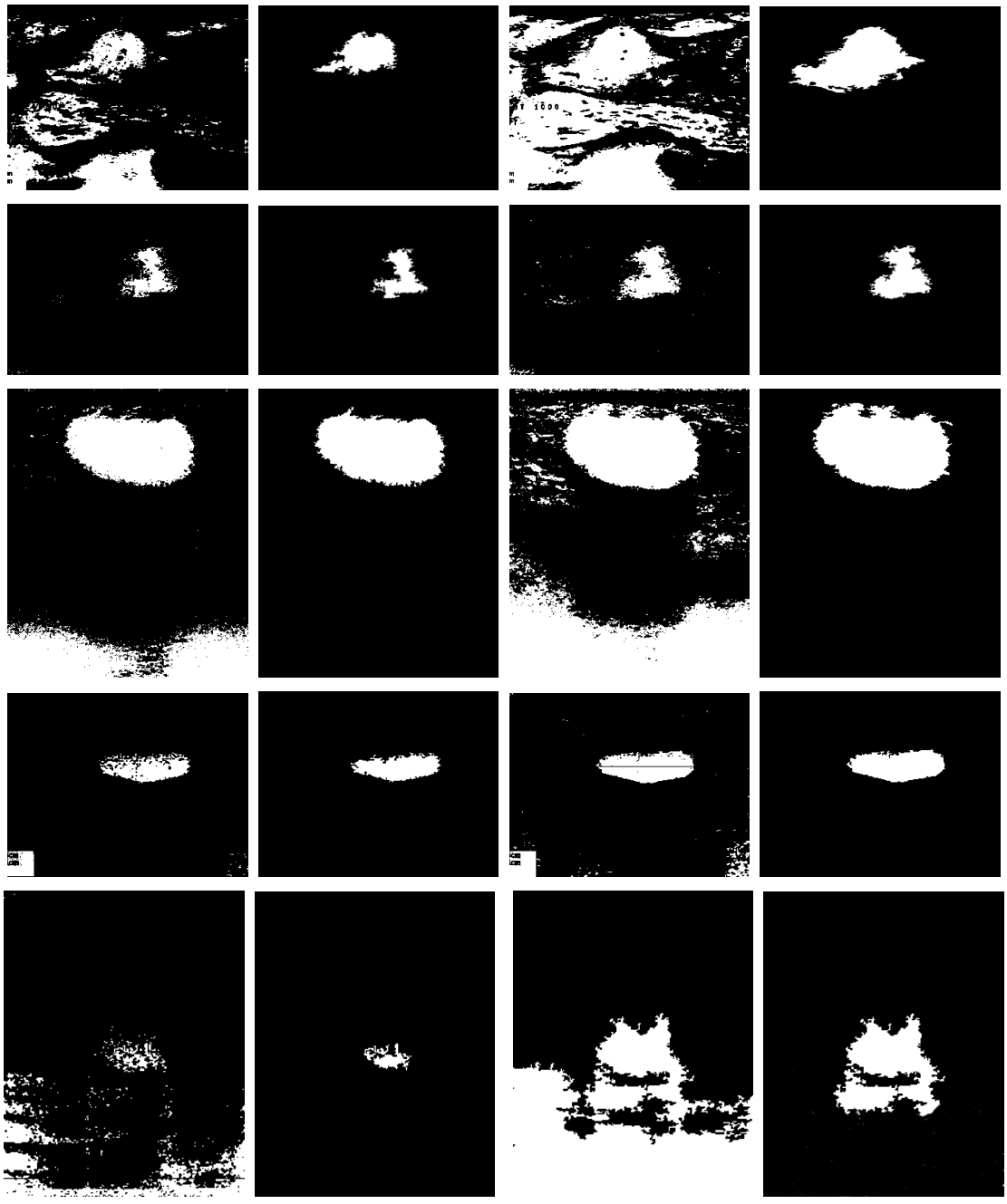


Figure 4.5: Area of Overlap measures of test case 1 in Table 4-1



(a) *(b)* *(c)* *(d)*

Figure 4.6: Images resulted from before and after cleaning stage. (a) and (b) show the before and after results of OFT, (c) and (d)demonstrate those of WOFT algorithm.

4.5 Comparison with well-known methods

Since Otsu and K-Means are two simple yet effective algorithms of image thresholding, we made a comparison between the results of these methods with OFT and WOFT algorithms. Otsu and K-Means methods are applied to both our synthetic and ultrasound image test cases and both accuracy and similarity measurements are calculated. Since selecting a center point has no effect on the result of these methods, the trial was conducted only once and the results are compared to the results of our test case 1 for both types of images. Tables 4-9 (area of overlap) and 4-10 (similarity) demonstrate the result and comparison for synthetic images and the same information but for ultrasound images trial are presented in Table 4-11 and Table 4-12.

Table 4-9 – Synthetic Images - Area of Overlap

	Average	STD
Otsu	56.31	5.10
K-Means	31.42	24.80
OFT	79.41	6.81
WOFT	90.31	5.07

Table 4-10 – Synthetic Images -Similarity

	Average	STD
Otsu	71.92	4.21
K-Means	42.33	30.65
OFT	73.95	8.01
WOFT	90.02	6.59

Table 4-11 – Ultrasound Images - Area of Overlap

	Average	STD
Otsu	13.12	8.02
K-Means	19.56	21.81
OFT	65.96	31.70
WOFT	77.56	19.47

Table 4-12 – Ultrasound Images - Similarity

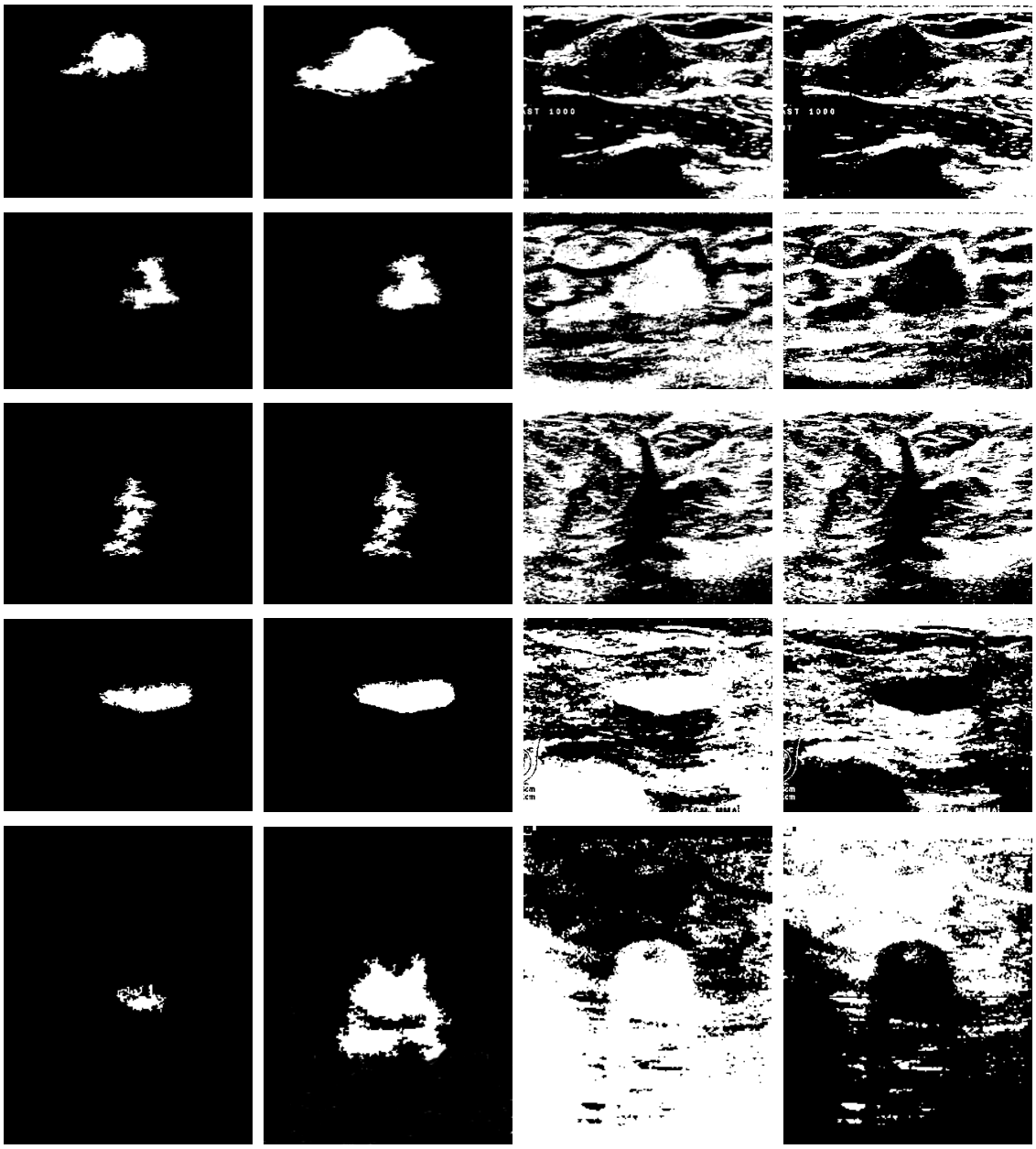
	Average	STD
Otsu	22.35	12.48
K-Means	28.73	23.41
OFT	71.57	32.91
WOFT	85.84	15.43

Analyzing the data provided in these tables, it is apparent that the proposed method surpasses both Otsu and K-Means methods in accuracy and similarity measures.

Some result images are provided for visual comparison in Figure 4.7. This figure illustrates the result of applying OFT and WOFT methods based on minimum entropy difference thresholding in comparison with resulted images from K-Means and Otsu methods.

Analyzing the results of before and after cleaning process values, we observed that a large percentage of low accuracy results occurred in cases that the image is cleaned. This fact confirms that the outcome of the image filtering concept is relative to the location of the sample point. Of course, this method would not be efficient if the sample point is not located inside the area of interest.

A more detailed analysis of the possible improvements of the current work and the future directions will be provided in the next chapter.



(a) *(b)* *(c)* *(d)*

Figure 4.7: Comparison of results with K-Means and Otsu methods. (a) and (b) are presenting the result of OFT and WOFT methods. (c) and (d) are resulted from K-Means and Otsu methods.

CHAPTER 5

Conclusion and Future Work

In this research, the initial motivation was to increase the accuracy of an existing image thresholding algorithm based upon introducing a weighted version of the parent algorithm. The weights are calculated based on some elementary characteristics of sub-images generating some encouraging outcomes.

The performance of the proposed method was verified by processing synthetic images along with breast ultrasound images as a real-world and challenging case.

As shown by the experimental results, it has been successfully demonstrated that the proposed WOFT algorithm can achieve more accurate results compared with the parent algorithm OFT, and with the benchmark algorithms Otsu and K-Means thresholding.

The future work should focus on further enhancements to the algorithm through different extensions such as learning the weights via an iterative procedure. Further, an optimal and comprehensive computer-aided image thresholding system must be able to classify the image without user interaction, an aspect that was not under consideration in this work. This requirement is generally desirable for industrial applications while many other applications, such as medical image analysis, may already benefit from the proposed semi-automated approach by providing some input from an experienced user.

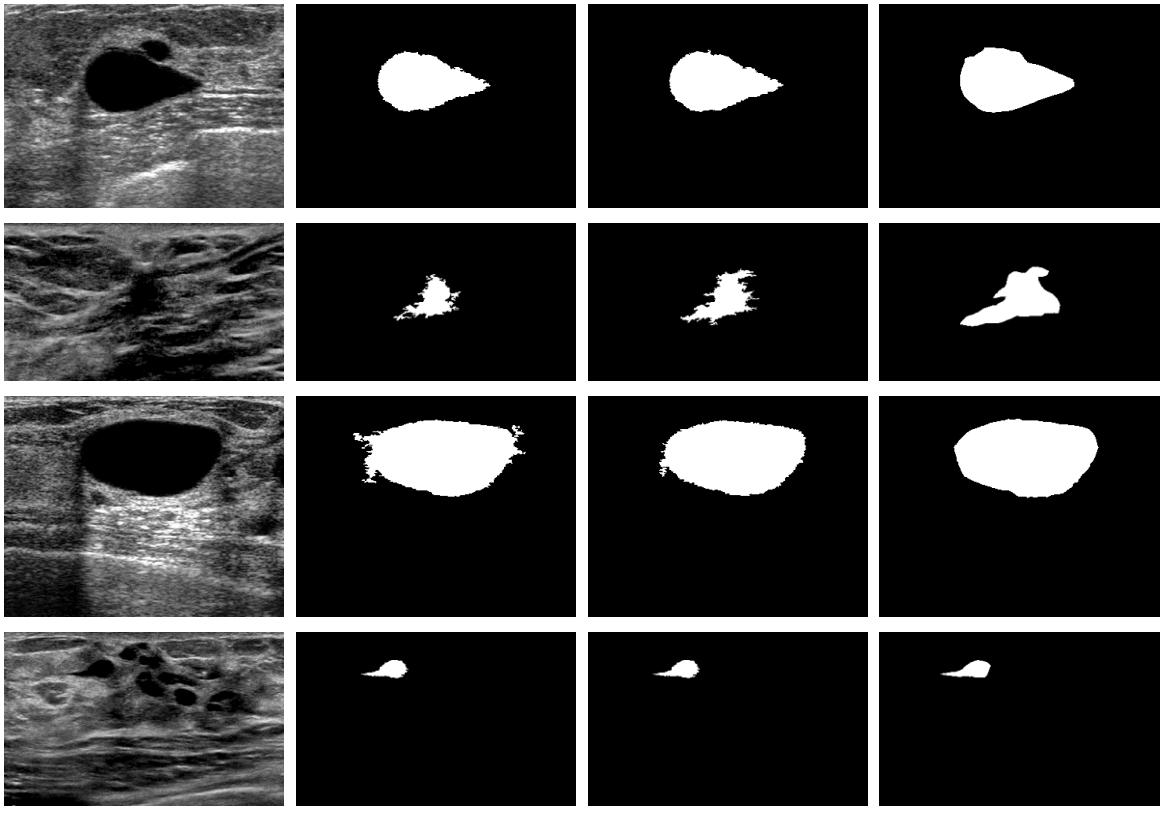
On the other hand, due to the logic applied to the filtering section of the algorithm, the object of the interest is not accurately thresholded if the central point is selected from outside of the object area. Modification to this part of the method can be considered as one of the other future directions.

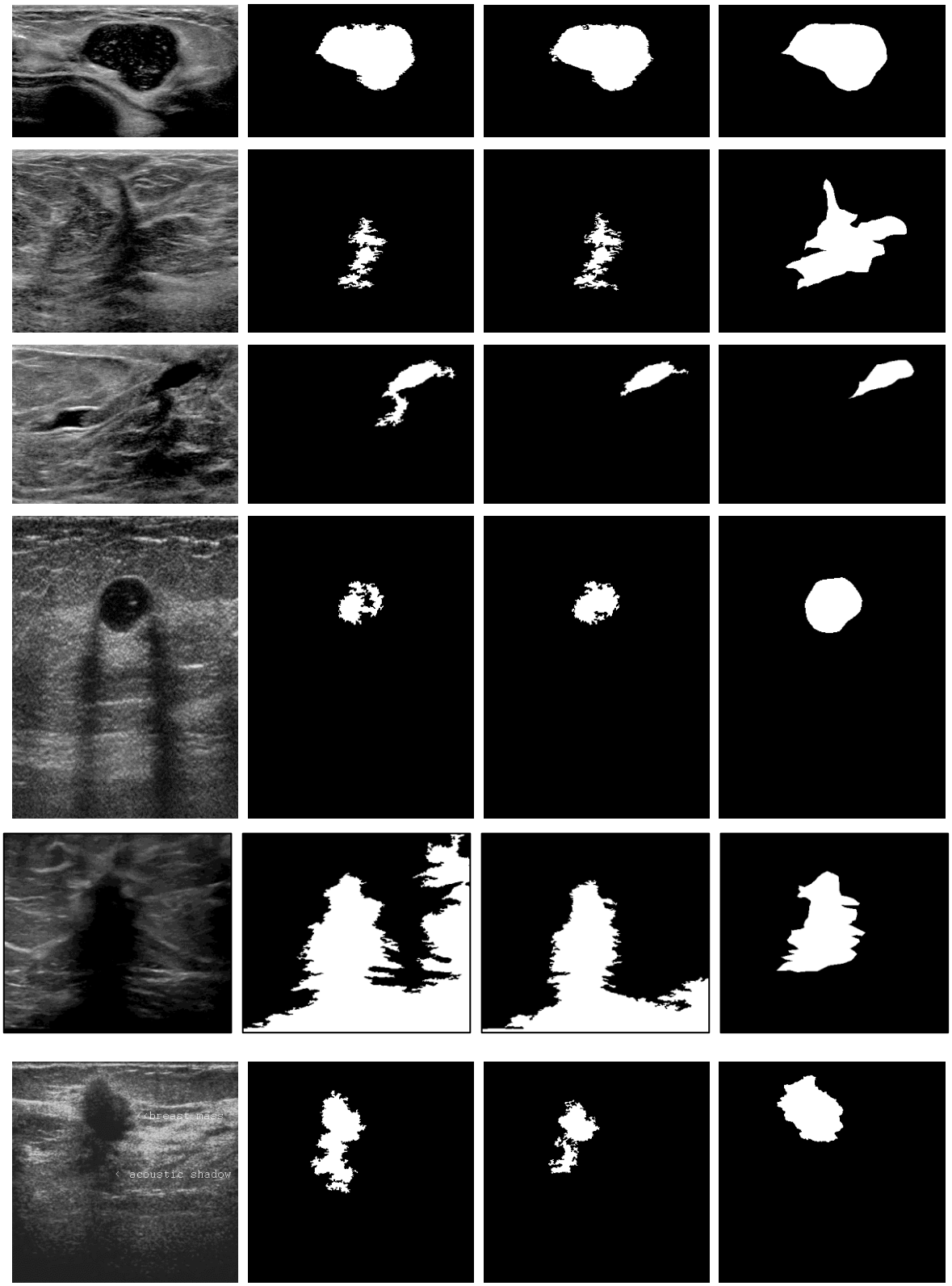
Better results can be achieved by enhancing image pre-processing stage. The possibilities can be further extended by the use of different noise removal and/or contrast enhancement techniques.

Even though we experimentally validated the robustness of the proposed thresholding technique using Type I Opposite Fuzzy Set, but should be considered just a preliminary investigation into applications of opposition in image thresholding/segmentation techniques. To continue the exploration, the applicability of Type II Opposite Fuzzy Sets should be studied and examined in future works as well.

Appendix A

More examples of applying OFT and WOFT methods (second and third column respectively) on ultrasound images are provided for visual comparison. Original and gold images are presented on the first and forth columns for each row.




breast mass
acoustic shadow

Bibliography

[1] Philips Image Library available at: http://www3.medical.philips.com/en-us/secure/images_site/index.asp?div=ultra

[2] http://www.cse.unsw.edu.au/~waleed/phd/html/node58.html

[3] W. N. Cottingham and D.A. Greenwood, "An Introduction to the Standard Model of Particle Physics," Cambridge University Press, Cambridge , 1998.

[4] D.A. Cruse, Lexical Semantics, Cambridge University Press, Cambridge, 1986.

[5] Lee R. Dice, "Measures of the Amount of Ecologic Association Between Species," Ecology 26 (3): 297–302, 1945.

[6] G. Gallo and S. Spinello, ''Thresholding and fast iso-contour extraction with fuzzy arithmetic,'' Pattern Recognition Letters, Vol. 21, pp. 31–44, 2000.

[7] C. A. Glasbey, "An analysis of histogram-based thresholding algorithms," Graphical Models and Image Processing, vol. 55, no. 6, pp. 532 – 537, 1993.

[8] R. Guo and S. M. Pandit, "Automatic threshold selection based on histogram modes and a discriminant criterion," Machine Vision and Applications, vol. 10, pp. 331–338, 1998.

[9] l.O. Hall, A. Bensaid, L.P. clarke, R. Velthuizen, M.S. Silbiger, and J.C. Bezdek, "A Comparison of Neural Network and Fuzzy Clustering Techniques in Segmenting Magnetic resonance Images of Brain", IEEE Transactions on Neural Network, 3,672-682, 1992.

[10] L.K. Huang and M.J.J. Wang, “Image Thresholding by Minimizing the Measures of Fuzziness”, Pattern Recognition, 28, 41-51, 1995.

[11] Paul Jaccard, "Etude comparative de la distribution orale dans une portion des Alpes et des Jura". In Bulletin del la Socit Vaudoise des Sciences Naturelles, volume 37, pages 547-579.

[12] C. V. Jawahar, P. K. Biswas, and A. K. Ray, ‘‘Investigations on fuzzy thresholding based on fuzzy clustering,’’ Pattern Recogn. 30(10) pp. 1605–1613, 1997.

[13] J. N. Kapur, P. K. Sahoo, and A. K. C. Wong, ‘‘A new method for gray-level picture thresholding using the entropy of the histogram,’’ Graph. Models Image Process. 29, pp. 273–285, 1985.

[14] R. L. Kirby and A. Rosenfeld, ‘‘A note on the use of (gray level, local average gray level) space as an aid in threshold selection,’’ IEEE Trans. Syst. Man Cybern. SMC-9, pp. 860–864, 1979.

[15] C. H. Li and C. K. Lee, ‘‘Minimum cross-entropy thresholding,’’ Pattern Recogn. 26, pp. 617–625, 1993.

[16] C. H. Li and P. K. S. Tam, ‘‘An iterative algorithm for minimum cross-entropy thresholding,’’ Pattern Recogn. Lett. 19, pp. 771–776, 1998.

[17] J. B. MacQueen, “Some Methods for classification and Analysis of Multivariate Observations,” Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297, 1967.

[18] A.R. Malisia and H. R. Tizhoosh, “Applying Opposition-Based Ideas to the Ant Colony System,” in Proc. IEEE Swarm Intelligence Symposium, Honolulu, HI, April 1-5, 2007, pp. 182-189.

[19] D. McLeish, “Monte Carlo Methods and Finance”, Wiley Chichester, 2005.

[20] L. O’Gorman, ‘‘Binarization and multithresholding of document images using connectivity,’’ Graph. Models Image Process. 56, pp. 494–506, 1994.

[21] N. Otsu, “A threshold selection method from gray level histograms,” IEEE Trans. Syst. Man Cybern. SMC-9, 62–66, 1979.

[22] S. K. Pal and A. Rosenfeld, ‘‘Image enhancement and thresholding by optimization of fuzzy compactness,’’ Pattern Recognition Letters, Vol. 7, pp. 77–86, 1988.

[23] J. M. Pearce, “Animal Learning and Cognition,” Psychology Press, 1997.

[24] S. Rahnamayan, H. R. Tizhoosh, and M. M. Salama, “Opposition-Based Differential Evolution Algorithms,” in Proc. IEEE Congress on Evolutionary Computation, Vancouver, July 16-21, 2006, pp. 7363-7370.

[25] S. Rahnamayan, H. R. Tizhoosh, and M. M. Salama, “Opposition-based Differential Evolution Algorithms for Optimization of Noisy Problems,” in Proc. IEEE Congress on Evolutionary Computation, Vancouver, July 16-21, 2006, pp. 6756-6763.

[26] S. Rahnamayan, H. R. Tizhoosh, and M. M. Salama, “A Novel Population Initialization Method for Accelerating Evolutionary Algorithms,” Computers and Mathematics with Applications, vol. 53, no. 10, pp. 1605-1614, 2007.

[27] S. Rahnamayan, H. R. Tizhoosh, and M. M. Salama, “Opposition-Based Differential Evolution (ODE) With Variable Jumping Rate,” in Proc. of IEEE Symposium on Foundations of Computational Intelligence (FOCI’07), Hawaii, April 1-5, 2007, pp. 81- 88.

[28] N. Ramesh, J. H. Yoo, and I. K. Sethi, ‘‘Thresholding based on histogram approximation,’’ IEE Proc. Vision Image Signal Process. 142(5), pp. 271–279, 1995.

[29] T. W. Ridler and S. Calvard, ‘‘Picture thresholding using an iterative selection method,’’ IEEE Trans. Syst. Man Cybern. SMC-8, pp. 630–632, 1978.

[30] A. Rosenfeld and P. De la Torre, ‘‘Histogram concavity analysis as an aid in threshold selection,’’ IEEE Trans. Syst. Man Cybern. SMC-13, pp. 231–235,1983.

[31] S. C. Sahasrabudhe and K. S. D. Gupta, ‘‘A valley-seeking threshold selection technique,’’ Comput. Vis. Image Underst. 56, pp. 55–65, 1992.

[32] F. Sahba and H.R. Tizhoosh, “Quasi-Global Oppositional Fuzzy Thresholding” in Proc. IEEE International Conference on Fuzzy Systems (FUZZ – IEEE), Korea, August 20-24, 2009, pp. 1346-1351.

[33] F. Sahba, "Reinforced Segmentation of Images Containing One Object of Interest," PhD Thesis, Department of Systems Design Engineering, University of Waterloo, 2007.

[34] Torsten Seemann, "Digital Image Processing using Local Segmentation," PhD dissertation, School of Computer Science and Software Engineering, Monash University, Australia, 2002.

[35] B. Sankur, M. Sezgin, "Survey over image thresholding techniques and quantitative performance evaluation," Journal of Electronic Imaging, 13, 1, 146 - 165, 2004.

[36] A. G. Shanbag, ''Utilization of information measure as a means of image thresholding,'' Comput. Vis. Graph. Image Process. 56, pp. 414–419, 1994.

[37] M. Shokri, H. R. Tizhoosh, and M. S. Kamel, "Opposition-Based Q_ Algorithm," in Proc. IEEE International Joint Conf. on Neural Networks (IJCNN), Vancouver, July 16-21, 2006, pp. 646-653.

[38] M. Shokri, H. R. Tizhoosh and M. S. Kamel, "Opposition-Based Q(lambda) with Non- Markovian Update," in Proc. IEEE Symposium on Approximate Dynamic Programming and Reinforcement Learning (ADPRL 2007), Hawaii, April 1-5, 2007, pp. 288-295.

[39] M. Shokri, "Oppositional Reinforcement Learning with Applications," PhD Thesis, Department of Systems Design Engineering, University of Waterloo, 2008.

[40] M. Shokri, H.R. Tizhoosh, "A Reinforcement Agent for Threshold Fusion," Journal of Applied Soft Computing, ASOC-309, Elsevier (Accepted 2006), 8, 1, 174-181, January 2008.

[41] M. Shokri, H.R. Tizhoosh, "Q(λ)-Based Image Thresholding," Canadian Conference on Computer and Robot Vision, pp. 504-508, 2004.

[42] H. R. Tizhoosh, "Opposition-Based Learning: A New Scheme for Machine Intelligence," in Proc. Int. Conf. on Computational Intelligence for Modelling Control and Automation - CIMCA'2005, Vienna, Austria, 2005, vol. I, pp. 695-701.

[43] H. R. Tizhoosh, "Reinforcement Learning Based on Actions and Opposite Actions," in Proc. Int. Conf. on Artificial Intelligence and Machine Learning, 2005.

[44] H. R. Tizhoosh, "Opposition-Based Reinforcement Learning," Journal of Advanced Computational Intelligence and Intelligence Informatics, vol. 10, no. 4, pp. 578-585, 2006.

[45] H. R. Tizhoosh, "Reinforcement Learning Based on Actions and Opposite Actions", in ICGST International Conference on Artificial Intelligence and Machine learning (AIML 2005), Cairo, Egypt.

[46] H.R. Tizhoosh, "Opposite Fuzzy Sets with Applications in Image Processing", in Proc. International Fuzzy Systems Association (IFSA) World Congress (IFSA-EUSFLAT 2009), Lisbon, Portugal, pp. 36-41.

[47] H. R. Tizhoosh and M. Ventresca, "Oppositional Concepts in Computational Intelligence". Springer, 2008.

[48] S. Venkatesh and P. L. Rosin, ''Dynamic threshold determination by local and global edge evaluation,'' CVGIP: Graph. Models Image Process. 57, pp. 146–160, 1995.

[49] M. Ventresca and H. R. Tizhoosh, "Improving the Convergence of Backpropagation by Opposite Transfer Functions," in Proc. IEEE International Joint Conf. on Neural Networks (IJCNN), Vancouver, July 16-21, 2006, pp. 9527-9534.

[50] M. Ventresca and H. R. Tizhoosh, "Opposite Transfer Functions and Backpropagation Through Time," in Proc. IEEE Symposium on Foundations of Computational Intelligence (FOCI'07), Hawaii, April 1-5, 2007, pp. 570-577.

[51] M. Ventresca and H. R. Tizhoosh, "Numerical Condition of Feedforward Networks with Opposite Transfer Functions," in Proc. IEEE International Joint Conf. on Neural Networks (IJCNN), Hong Kong, June 1-8, 2008, pp. 3233-3240.

[52] O. Virmajoki and P. Franti, "Fast pairwise nearest neighbor based algorithm for multilevel thresholding," Journal of Electronic Imaging, vol. 12(4), pp. 648–659, Oct. 2003.

[53] H. Yan, "Unified Formulation of a Class of Image Thresholding Techniques," Pattern Recognition, 29, 12, 2025-2032, 1996.

[54] Zuiderveld, Karel. "Contrast Limited Adaptive Histograph Equalization," Graphic Gems IV. San Diego: Academic Press Professional, pp. 474–485, 1994.