THICKNESS ESTIMATION, AUTOMATED CLASSIFICATION AND NOVELTY DETECTION IN ULTRASOUND IMAGES OF THE PLANTAR FASCIA TISSUES

Abdelhafid Boussouar

Informatics Research Centre
School of Computing, Science and Engineering
University of Salford, UK

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS OF
THE DOCTOR OF PHILOSOPHY, MARCH 2019

Contents

List of Tables

List of Figures

Acknowledgements

I would like to express my sincere gratitude to Prof. Farid Meziane for his continuous critical advice, guidance, encouragement and support throughout this project. Despite his busy schedule, he was always available for help, advice and guidance.

I would also like to thank my co-supervisor Dr. Gillian Crofts and all the research participants from the Health Sciences Department for their help and crucial assistance, particularly, during the data collection and system testing phases. Without their valuable opinions and ideas on the proposed system, this thesis would not have been completed.

Finally, I would like to express my gratitude and appreciation to my parents, my wife, my children, my sisters, my brothers, and my best friends for their love, patience, encouragement, constant support and understanding during the course of this project.

Declaration

No portion of the work referred to in this thesis has been submitted in support of an application for another degree or qualification of this or any other university or other institute of learning.

Abstract

The plantar fascia (PF) tissue plays an important role in the movement and the stability of the foot during walking and running. Thus it is possible for the overuse and the associated medical problems to cause injuries and some severe common diseases. Ultrasound (US) imaging offers significant potential in diagnosis of PF injuries and monitoring treatments. Despite the advantages of US, the generated PF images are difficult to interpret during medical assessment. This is partly due to the size and position of the PF in relation to the adjacent tissues. This limits the use of US in clinical practice and therefore impacts on patient services for what is a common problem and a major cause of foot pain and discomfort. It is therefore a requirement to devise an automated system that allows better and easier interpretation of PF US images during diagnosis. This study is concerned with developing a computer-based system using a combination of medical image processing techniques whereby different PF US images can be visually improved, segmented, analysed and classified as normal or abnormal, so as to provide more information to the doctors and the clinical treatment department for early diagnosis and the detection of the PF associated medical problems. More specifically, this study is required to investigate the possibility of a proposed model for localizing and estimating the PF thickness a cross three different sections (rearfoot, midfoot and forefoot) using a supervised ANN segmentation technique. The segmentation method uses RBF artificial neural network module in order to classify small overlapping patches into PF and non-PF tissue. Feature selection technique was performed as a post-processing step for feature extraction to reduce the number of the extracted features. Then the trained RBF-ANN is used to segment the desired PF region. The PF thickness was calculated using two different methods: distance transformation and a proposed area-length calculation algorithm. Additionally, different machine learning approaches were investigated and applied to the segmented PF region in order to distinguish between symptomatic and asymptomatic PF subjects using the best normalized and selected feature set. This aims to facilitate the characterization and the classification of the PF area for the identification of patients with inferior heel pain at risk of plantar fasciitis. Finally, a novelty detection framework for detecting the symptomatic PF samples (with plantar fasciitis disorder) using only asymptomatic samples is proposed. This model implies the following: feature analysis, building a normality model by training the one-class SVDD classifier using only asymptomatic PF training datasets, and computing novelty scores using the trained SVDD classifier, training and testing asymptomatic datasets, and testing symptomatic datasets of the PF dataset. The performance evaluation results showed that the proposed approaches used in this study obtained favourable results compared to other methods reported in the literature.

Chapter 1

Introduction

1.1 Motivation of the study

The plantar fascia (PF) or plantar aponeurosis is an aponeurotic (i.e. binding muscles together or connecting muscles to bones) thick, fibrous and strong connective tissue. It provides stability to the medial longitudinal arch of the foot (Huang et al., 1993). It originates at the medial calcaneal tuberosity (i.e. the posterior extremity of the calcaneus, forming the projection of the heel) and extends towards the digits in three different structural bands: medial, central, and lateral (Chang, 2010) (Figure 1.1). The central area is the largest, most affected by disease and most susceptible to deformities (Kwong et al., 1988; Kelikian, 2012).

The PF plays an important role in stabilizing the foot during walking and running. However, a commonly encountered condition is foot pain due to overuse. Foot pain may be due to a number of causes such as: plantar fasciitis (Pfeffer et al., 1999), traumatic disorders (e.g. PF rupture) and plantar fibroma or fibromatosis (the existing of a fibrous nodule (or multi-nodules) in the arch of the foot) (McNally and Shetty, 2010). Plantar fasciitis is considered the most frequent cause of heel pain in the UK (it approximately accounts for about 80% of heel pain cases) and it is more likely to affect people who are overweight and those with active participation in sports (Akfirat et al., 2003; Beeson, 2014). It has been estimated that 10% of the general population are affected during their middle age years (commonly in 40-60 years of age) (Chang, 2010; Zhiyun et al., 2013; Neufeld and Cerrato, 2008). The incidence of the plantar fasciitis condition makes up

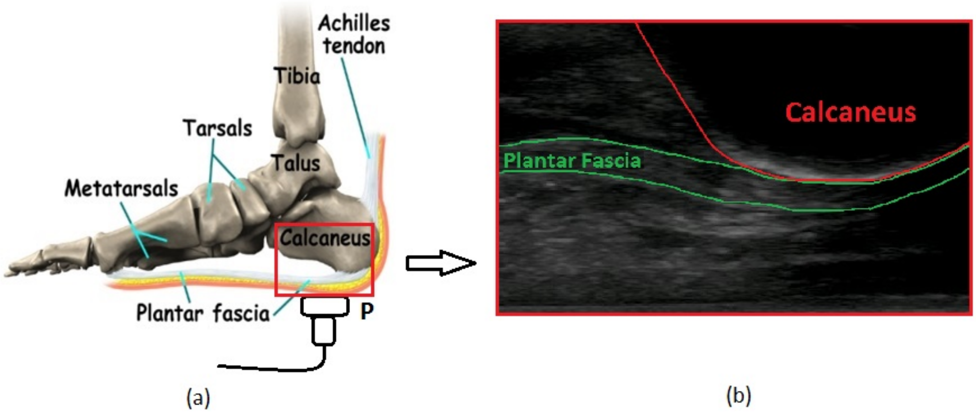


Figure 1.1: Plantar fascia region: (a) Anatomical illustration diagram showing the anatomical location of the plantar fascia and positioning of the ultrasound probe, P. (b) A longitudinal sonogram of the scanned region related to (a), showing the plantar fascia area and the calcaneus

25% of all injuries in runners (Clement et al., 1981) and 8% of all injuries in athletes (Landorf et al., 2006).

Assessment of foot pain typically involves clinical examination and diagnostic imaging (Park et al., 2014). The role of diagnostic imaging is to provide objective information which significantly informs clinical decisions on treatment options. Both ultrasound (US) and magnetic resonance imaging (MRI) are non-invasive imaging modalities with considerable potential for the diagnosis and monitoring of a wide range of PF medical problems (Buchbinder, 2004; Puttaswamaiah and Chandran, 2007; Shazia et al., 2011; McPoil et al., 2008). However, MRI is expensive, with limited accessibility and not appropriate for the majority of clinically based research studies, especially where frequent diagnosis is needed. By contrast, US imaging is a real-time imaging technique which is readily available, fast, causes no radiation exposure, portable, accurate, and cost-effective (Pope, 1999; Szabo, 2013). Moreover, it is considered to be highly reliable and favourable in the diagnosis of diabetic foot with plantar fasciitis, ankle infections and damaged soft tissue (Crofts et al., 2014; Angin et al., 2014; Szabo, 2013; Akfirat et al., 2003).

Although US imaging offers many advantages in the diagnosis of PF, it is often seen as being operator dependent when used by non-expert users. This means imaging results are often dependent on the expertise of the operator to acquire high quality images. In addition, the quality of images can also be affected by the presence of speckle noise.

This is a type of multiplicative noise which occurs during the process of acquisition and transmission (Ganzalez and Woods, 2002). The presence of speckle noise in US images reduces image contrast, thereby limiting the detection of small, low-contrast lesions (Burckhardt, 1978), such as those in the PF US images. The effect of speckle noise in PF US images is to destroy or diffuse the PF edges, making medical interpretation and biometric measurements challenging, and therefore impacting the accuracy of diagnosis. In addition, speckles limit the efficiency of some US image processing applications, such as automated segmentation, feature extraction, image registration and novelty detection.

Research studies have reported thickening, biconvexity and hypoechoic deformities of the PF as part of the diagnostic criteria and characteristic features of PF (Park et al., 2014). Increased thickness of the plantar fascia with values of $>$ 4 mm (Figure 1.2b) and decreased echogenicity of plantar fascia are considered as symptomatic cases (Fabrikant and Park, 2011; Wearing et al., 2007; Saber et al., 2012). Rupture, intratendinous calcification and perifascial oedema are also considered during US diagnosis (Akfirat et al., 2003).

Figure 1.2 shows normal and symptomatic plantar fascia tissue where thickness is very significantly changed in symptomatic cases.

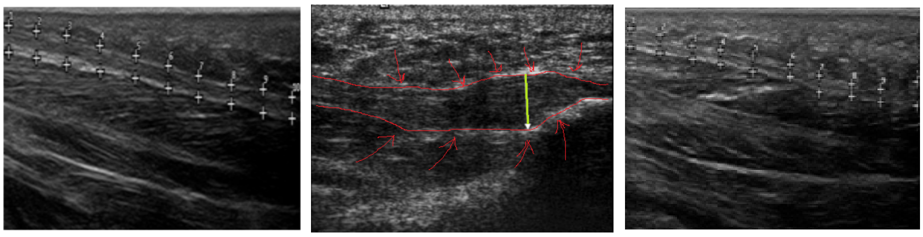


Figure 1.2: PF US images,(a) Normal PF US image, (b-c) Symptomatic PF US images: (b) Hypertrophy (Swelling), (c) Atrophy (Narrowing)

There are different protocols in the literature used to measure the PF thickness:

- Measurement of the plantar fascia has primarily been limited to thickness at the calcaneous insertion site by either inter- or intra-rater reliability (Cheng et al., 2012);

- Measuring the PF by the average bias of repeated measurements (Wearing et al., 2004);
- Some recent works (Crofts et al., 2014) have shown that the thickness of the plantar fascia varies along its length. Therefore, a reliable means of quantifying PF thickness in the rearfoot, midfoot and forefoot structures could be advantageous (Figure 1.3).

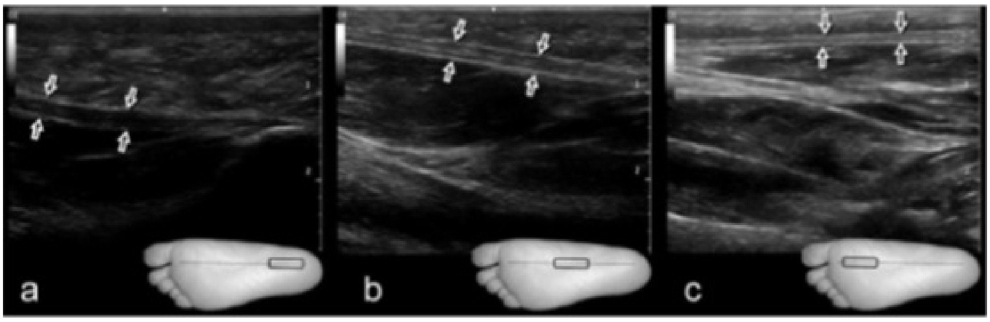


Figure 1.3: Probe position, longitudinal orientation and sample US images for all PF different structures. a) Rearfoot PF section; b) Midfoot PF section; and c) Forefoot PF section.

1.2 Problem definition and challenges

Available literature has demonstrated that US imaging is reliable in foot and ankle assessment and offers a real-time effective imaging technique that is able to reliably confirm structural changes, such as thickening in the internal echo structure associated with diseased or damaged tissue. However, the generated PF US images are difficult to interpret during medical assessment. To date, no automatic methods (e.g. automated computer aided detection systems (CAD)) have been developed in the field of PF segmentation, classification and novelty detection, because PF has been treated as a misunderstood condition with a little agreement regarding the perfect related treatment solutions. The current treatments mainly based on the non-proof or decision making based treatments (e.g. calf stretches, lateral rotator strengthening and footwear). Additionally, most outlined treatment results depend on anecdotal experience or integrations of various modalities (90% non-surgical treatments of PF disorders and only 10% surgical treatment interventions) (Neufeld and Cerrato, 2008; Miller and Latt, 2015). This makes the task of segmenting PF US images and classifying PF subjects into normal and abnormal in the clinic more challenging. However, this also presents an opportunity for developing novel methods to facilitate this task for clinicians. Four major issues should be overcome in applying the proposed medical image processing techniques to PF US image datasets:

1) **Speckled and low contrast PF US images**
   The generated PF US images are usually affected by multiplicative speckle noise and low resolution in the process of acquisition and transmission, making the visual interpretation and measurement more complicated during diagnosis. This will also alter the efficiency of US image processing applications.

2) **Hard to delineate and measure the PF in different cites**
   The anatomical substructure of the foot soft tissues, the size and position of the PF in relation to the adjacent tissues make the delineation of the PF area and estimating its thickness challenging. Additionally, as demonstrated in some recent works (Crofts et al., 2014), the multi-site nature of PF region (rearfoot, midfoot and forefoot) poses a significant challenge in calculating PF thickness in different sites.

3) **Lack of class separation between all PF subjects**
   Finding the class separation between asymptomatic (normal) and symptomatic (abnormal) ultrasound images of the plantar fascia is a challenging process in characterizing and classifying different PF subjects.

4) **Imbalanced dataset problem and detecting PF abnormalities embedded in normal datasets**
   The small number and unbalanced in the proportion of available PF US images (only 250 normal and 30 abnormal samples in the collected data) can cause challenges for training and testing the proposed system, with specific difficulties in normal classification process. In addition, finding the PF normality class outliers is a challenging process in detecting PF abnormalities behaviour using different PF subjects.

1.3 Aim and objectives

The purpose of this study is to develop and evaluate an automatic computer-based system using a combination of medical image processing techniques such as, despeckle filtering, feature extraction, segmentation, artificial neural networks classification and abnormalities detection techniques, whereby the PF US images can be visually improved and analysed, so as to provide more visual information to the clinicians for reliable assessment and diagnosis. More specifically, this study attempted to segment, measure, classify plantar fascia US images into normal and abnormal images and to develop a suitable novelty detection (one-class classification) approach for PF US image dataset based primarily on normal PF data and a set of abnormal testing data. This aim was addressed according to the following objectives:

- To investigate and evaluate image processing techniques such as: speckle noise reduction, automatic segmentation, thickness estimation, feature extraction, feature ranking, feature selection, supervised classification, and novelty (i.e. abnormality) detection (one-class classification or outlier classification) approaches applied to medical US images.

- To develop and evaluate a medical ultrasound image processing system where the PF US images can be visually improved, segmented, measured, classified, analysed using novelty detection methods, and used for medical diagnosis.

- To assess the clinical validation of the proposed system using real dataset (Symptomatic and Asymptomatic PF US images), this study involves setting, recruitment of voluntary research patients participants, US radiologist and pathologist experts from the Health Sciences Department, University of Salford.

1.4 Research contributions

In this thesis, four main contributions to knowledge have emerged that fall into the domain of medical US image processing such as acoustic speckle noise reduction, PF segmentation and thickness estimation, PF characterization and classification and PF anomaly detection:

1. The careful selection of speckle reduction technique is very significant in the despeckling of the PF US images and improving its usefulness after the effect of the speckle noise phenomenon during image acquisition and transmission. For this aim, a comparative evaluation study has been made to analyse the qualitative and quantitative ability of some existing despeckling techniques in the assessment of 2-D PF US images. For this study, a medical imaging MatLab GUI frame work was developed. This frame work supports a wide range of US image speckle reduction functionalities for the evaluation of seven despeckling groups (a total of 16 filters). Additionally, the trade-off between the acoustic speckle phenomenon reduction and PF US image detail preservation was assessed using 11 image quality measures; feature ranking and selection analysis; and visual evaluation by two medical experts. The results of this study have proved that the filtering methods based on dual tree complex wavelet transform (DT-CWT) using BayesShrink subband thresholding and different thresholding functions namely, soft, hard, trimmed and bivariate ((DT-CWT_S, DT-CWT_H, DT-CWT_T and DT-CWT_B)) achieved the best results. These filters have proved: the ability to reduce speckle noise effectively while preserving the edges and details of the PF US images; the ability to improve feature ranking and selection assessment, and the ability to visually improve the PF US images. The advantage of these methods have also demonstrated that they are important mathematical tool which can have a great potential for PF US imaging segmentation, features extraction, selection and classification.

2. A novel automatic segmentation approach which for the first time extracts ultrasound data to estimate size across three sites of the PF (rearfoot, midfoot and forefoot). For this purpose, a medical image processing MatLab application and GUI frame work was implemented. This frame work supports a wide range of medical US image functionalities such as speckle noise reduction, PF delineation, PF thickness calculation, feature extraction and selection, classification and novelty detection as shown in Appendix C. This segmentation approach uses radial basis function artificial neural network module (RBF-ANN) in order to classify small overlapping patches as belonging or not-belonging to the region of interest (ROI) of the PF tissue. Feature ranking and selection techniques were also used as

a post-processing step for feature extraction to reduce the dimension and redundancy of the extracted features. The trained RBF-ANN classifies the PF image overlapping patches into PF and non-PF tissues, and then it is used to segment the desired PF region in three PF sites (reafoot, midfoot and forefoot). Furthermore, two different methods were proposed to estimate the thickness of the PF region such as distance transformation and area-length calculation. The statistical analysis results demonstrated that the area-length thickness estimation approach overcomes the distance transformation approach in terms of high significant positive pairing between the manual thickness estimation and the automatic assessment. In addition, the results have also showed that there is a clear difference between different PF sites and the thicknesses of PF vary along the length of the foot. It also obtained favourable results compared to other active contour methods (reported in the literature) which are mainly based on the variational information of grayscale intensities of the image, and they performed poorly when there was no much difference between the foreground and background means, especially in US images. The proposed segmentation approach is very helpful to assist the physicians and doctors for early PF problems diagnosis. This also reduce the time required by physicians for PF pathology diagnosis and the subjectivity that accompanies manual delineations and PF thickness estimation. The effectiveness of the proposed method supports the potential of its use in US imaging and other US image processing applications including feature extraction, feature classification and novelty detection.

3. An automated supervised classification approach to distinguish between normal and abnormal (plantar fasciitis) PF subjects. This will also facilitate the characterization and the classification of the PF region for the diagnosis of patients with inferior heel pain at risk of plantar fasciitis. Six feature measure sets (a total of 42 features) were extracted from the PF ROI segments. In addition, features normalization, features selection and ranking (to rank and select the extracted features, based on their weights importance) using an unsupervised infinity selection method have also been introduced for the characterisation and the classification

of normal and abnormal PF samples. In the characterisation of normal and abnormal US PF subjects only the top 28 feature sets were selected. The F-score measure was used to select the best features for six selected classifiers (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN) using different selected feature sets (1-40) and 10-folded cross-validation technique. The results of this study have demonstrated that the RBF-NN module overcomes other classification techniques and achieved its best in terms of low misclassified PF instances, high performance measures for Recall, F-Score, MCC and AUC and average measure, with low execution cost. This also has demonstrated the advantage of the RBF-NN module when used in the classification and the characterization of different US PF samples.

4. A proposed novelty (abnormality) detection approach (i.e. one-class classification model using only normal dataset during training phase) for the PF ultrasound images datasets, based primarily on the normal data. This model uses support vector data description (SVDD) for one-class classification task to tackle the problem of imbalanced datasets and to draw PF dataset normality outliers and define abnormalities behaviour (i.e. identifying novelty scores or thresholds that separate normal and abnormal classes using only normal). These scores were computed using only the normal training PF datasets. For the testing and the evaluation phase, both normal and abnormal PF datasets were used. The optimal threshold is set by the validation data. In order to select the top features for each anomaly detection classifier (Parzen, GMM, GPOC, SOM and SVDD), G-mean measure was computed using the top 28 feature sets. The SVDD achieved his best using the top 16 features with G-mean value of 0.873. The results of this study have proven the effectiveness of SVDD classifier when compared to other selected modules (Parzen, GMM, GPOC and SOM) in terms of high performance measures (B-Accuracy, F-score, MCC percent, Gmean, ROC plot AUC) and low time complexity. This also facilitate the possibility of detecting early warning of plantar fasciitis or other PF anomalies using novelty scores estimation approach.

1.5 Scope and limitations of the study

In this study different medical image processing solutions are proposed for (only 2D normal and abnormal) PF ultrasound image analysis and interpretation. Different approaches that combine the advantages of intelligent techniques are presented such as ANN special theory based segmentation technique (i.e. an automatic segmentation approach derived from other domain of knowledge such as Radial Basis Function neural network (RBF-ANN) pattern recognition and classification), supervised classification and novelty detection techniques in conjunction with different feature extraction and selection methods. A number of limitations were unintentionally imposed on the present study such as:

1. This study was conducted only on small size of US images dataset especially abnormal cases (36 images) due to the lack of patients' participation in the study and the time limit for data collection, this will limit the performance estimates and the results generalization. The quantity of 286 (250 normal and 36 abnormal) 2D ultrasound images is quite low especially for a quantitative comparison of several methods. Because, it minimizes the power of the study and maximizes the margin of the error leading to false-positive results, or they over-estimate the magnitude of the results. Additionally, this will also affect the machine learning process when building a classification model. Consequently, increasing the US dataset size increases the confidence level of our research study and decreases the margin of error (e.g. the standard deviation). Therefore, the extension of the implemented approaches for using datasets augmentation methods to artificially increase the PF US sample size is the topic of future work.

2. The construction of the ground truth used to assess the performance of the proposed approaches. Indeed, we used manual segmentation and measurement performed by just two physician experts directly on ultrasound images, acquired according to a precise protocol. The datasets generated by the two experts were used to establish the ground truth values of the PF segmentation and thickness estimation. So, using manual segmentation by a only two operators reduces the significance of the performance assessment and vice versa.

3. The manual segmentation and analysis of the large PF US datasets is a tedious,

time-consuming and complex task for physicians and clinicians, who have to manually select the ROIs and extract useful diagnostic information. This analysis will lead to inter- or intra-operator variability errors.

4. Some of the acquired PF US images are poor quality images (may be due to incorrect ultrasound machines settings, incorrect probe focus, or wrong protocol, etc.), this will limit and alter the efficiency of US image processing applications; especially, speckle noise reduction results, feature extraction and selection, visual evaluation by medical experts, and supervised classification which needs a lot of good samples from each PF class when training the classifiers. This could also alter the efficiency of PF diagnosis and the effect of the treatment.

5. Some of the proposed approaches using supervised classification require new training each time whenever there is an increase or change in PF US image database. Furthermore, we need to select the best PF samples from each PF class while we are training all our machine learning classifiers. Consequently, this would be really challenging when considering the classification of huge PF US datasets that needs a lot of computation time for training phase. Therefore, the extension of the implemented machine learning approaches for using other methods such as unsupervised classification techniques should be considered as a future work.

6. This study operates only in 2D US image domain for processing normal and abnormal US images, this will affect other information estimation, such as broader location, volumes, texture context, etc. The only way of conveying all these useful information by using 3D PF US representation (using many 2D PF US slices with a sampling frequencies). Additionally, most physicians prefer a 3d representation of the medical images for diagnosis (Luboz et al., 2014), because it appears to be closer to the real world. Therefore, the extension of the implemented approaches for processing other pathological PF tissues (e.g. rupture, fibromatosis, fibroma) in 3D US and MRI image domain is the topic of future research.

1.6 Outline of the thesis

Chapter 1 introduces the motivation for this study, a brief description of the problem and the five major challenges which must be overcome in this thesis, followed by the aim and key objectives of the thesis. It also provides a structure of the research undertaken and an outline of the thesis.

Chapter 2 is a literature review that describes medical image processing techniques used in US images with their limitations. First, it presents an overview of the ultrasound imaging process using a pulse-echo technique for clinical real-time imaging. Secondly, it introduces acoustic speckle noise phenomena, nature, source and generalization. Thirdly, it discuses the three-step medical ultrasound image processing model including image acquisition, image processing and analysis, and image interpretation. Finally, the chapter presents different medical ultrasound image processing techniques such as speckle noise reduction techniques, image enhancement techniques, segmentation techniques, feature extraction and selection techniques, classification techniques and novelty detection techniques along with their mathematical background, advantages and shortcomings. This chapter also summarises the relevant medical processing approaches and derives the most appropriate techniques that can be used in this study.

Chapter 3 introduces the comparative evaluation study of some selected existing speckle-reducing filtering methods (7 groups and 16 filters) based on image quality evaluation metrics; feature extraction, ranking and selection analysis; and visual evaluation by two clinical experts in the assessment of 286 normal and abnormal plantar fascia US images along with experimental findings and discussions.

Chapter 4 presents the plantar fascia segmentation and thickness estimation study with the finding results, evaluation analysis and discussions. In particular, this chapter shows how artificial neural network (using radial basis function classifier) can be applied to plantar fascia tissue analysis in medical ultrasound imagery. It includes methods to automatically segment the plantar fascia tissue in different sites (rearfoot, midfoot and forefoot) and estimate its thickness.

Chapter 5 introduces automatic supervised classification study to identify and classify normal, abnormal plantar fascia subjects (285 images) using different sets of features (42 features), features ranking and selection methods (for features dimensionality reduction and selection) with the experimental outcomes, analysis and discussions.

Chapter 6 includes the plantar fascia novelty detection model using SVDD (one-class classification approach) applied to the normal plantar fascia ultrasound images dataset (with insufficient abnormal datasets) for defining plantar fascia abnormalities behaviour (novelty or abnormality threshold). Different novelty models have been also investigated for novelty detection in the plantar fascia images datasets considered in this thesis including the experimental results, discussions and conclusions.

Chapter 7 gives a general summary of the presented research, draws contribution and conclusions from the thesis, and discusses some possible directions and guidance on the scope for further future work.

1.7 Journal Publications Resulting from this Research

Published Papers

- Boussouar, A., Meziane, F., Crofts, G., 2017a. Plantar fascia segmentation and thickness estimation in ultrasound images. Computerized Medical Imaging and Graphics 56, 60–73

Under Review Papers

- Boussouar, A., Meziane, F., Hogg, P., Hashmi, F., 2017b. Speckle noise reduction in ultrasound imaging of the plantar fascia, a comparative evaluation. Submitted to ACM Computing Surveys (Second round review)

- Boussouar, A., Meziane, F., 2018c. Plantar fascia characterization and classification based on machine learning techniques for ultrasound images. Submitted to Springer Neural Computing and Applications

Working Papers

- Boussouar, A., Meziane, F., 2018b. Novelty detection for ultrasound images of the plantar fascia
- Boussouar, A., Meziane, F., 2018a. Computer-based medical ultrasound image processing system and methods

1.8 Thesis structure illustration showing how the chapters are connected to each other

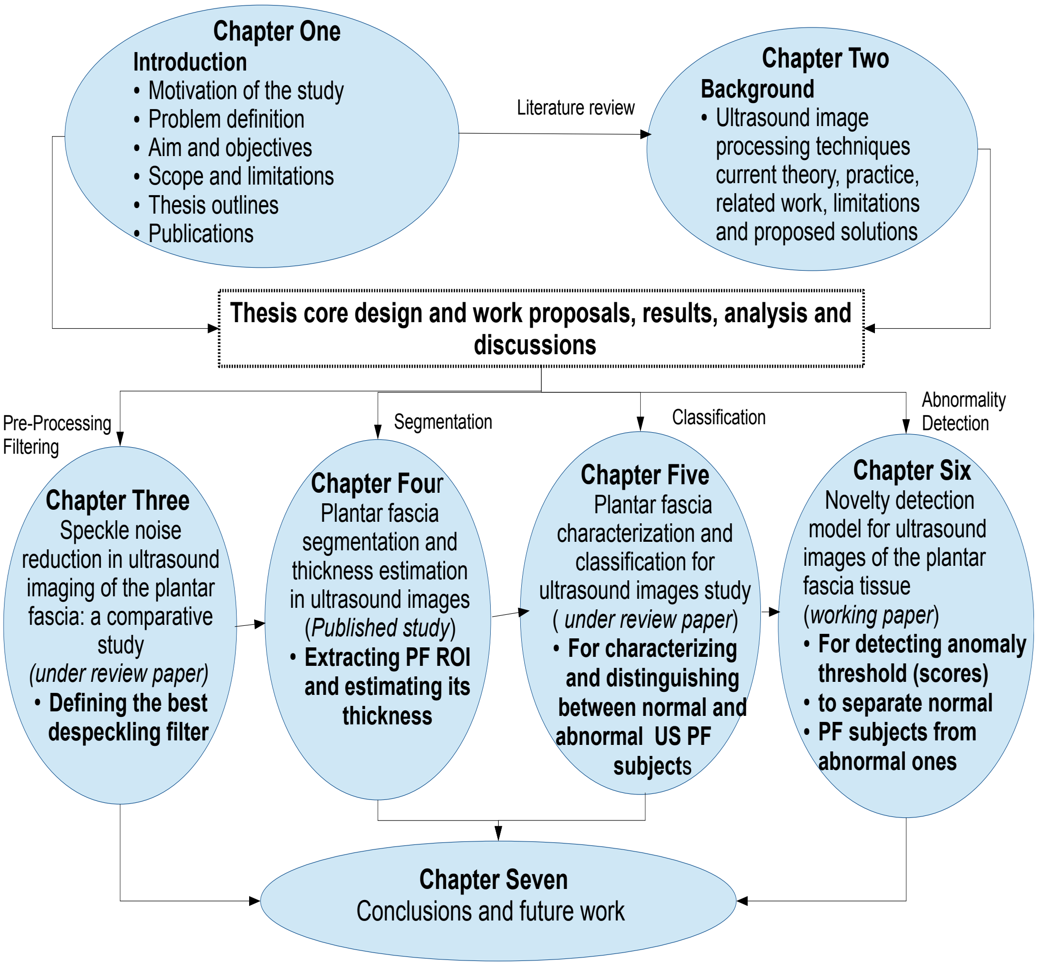

Chapter One
Introduction
• Motivation of the study
• Problem definition
• Aim and objectives
• Scope and limitations
• Thesis outlines
• Publications
Literature review
Chapter Two
Background
• Ultrasound image processing techniques current theory, practice, related work, limitations and proposed solutions
Thesis core design and work proposals, results, analysis and discussions
Pre-Processing Filtering
Segmentation
Classification
Abnormality Detection
Chapter Three
Speckle noise reduction in ultrasound imaging of the plantar fascia: a comparative study
(under review paper)
• Defining the best despeckling filter
Chapter Four
Plantar fascia segmentation and thickness estimation in ultrasound images
(Published study)
• Extracting PF ROI and estimating its thickness
Chapter Five
Plantar fascia characterization and classification for ultrasound images study
(under review paper)
• For characterizing and distinguishing between normal and abnormal US PF subjects
Chapter Six
Novelty detection model for ultrasound images of the plantar fascia tissue
(working paper)
• For detecting anomaly threshold (scores)
• to separate normal
• PF subjects from abnormal ones
Chapter Seven
Conclusions and future work

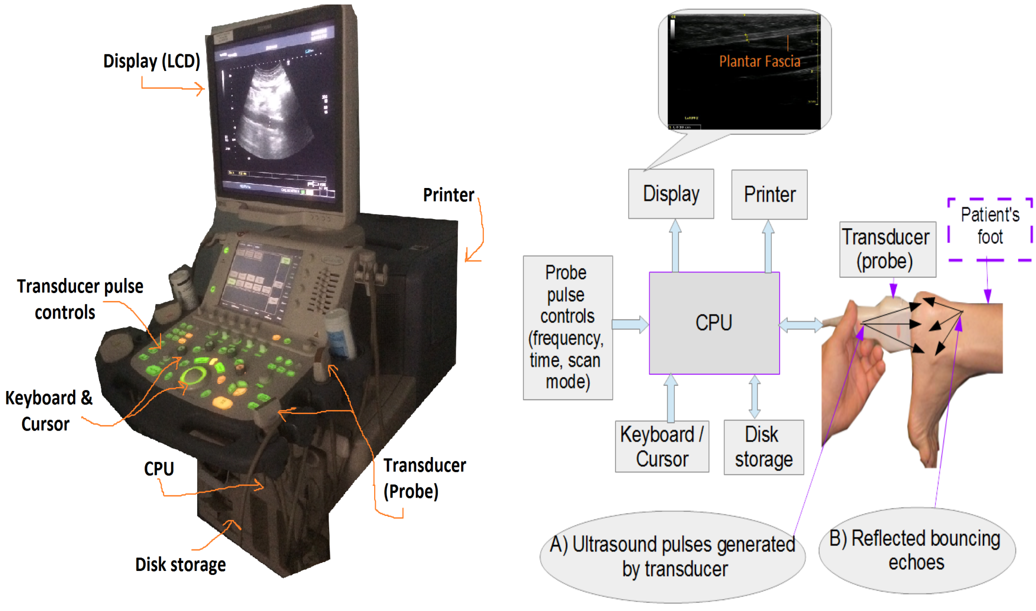
Figure 1.4: The whole thesis structure showing how all chapters are connected to each other

Chapter 2

Ultrasound Image Processing Techniques: Background and Literature Review

2.1 Ultrasound imaging

During the PF medical assessment, the ultrasound (US) imaging process involves: (1) the use of a high specification ultrasound or ultrasonography machine with different transducers (Figure 2.1 (a)), (2) a clinical radiologist who acquires the images, and (3) a physician who subsequently interprets the acquired images and makes the required diagnosis. Ultrasound imaging is a non-invasive imaging modality with considerable potential for the diagnosis and monitoring of a wide range of medical problems. Compared to other imaging techniques, including X-ray imaging, computerised tomography (CT), and magnetic resonance imaging (MRI), US imaging has been shown to be a safe (non-ionising), real-time imaging technique that is readily available, portable, accurate, cost-effective (Pope, 1999; Szabo, 2013), easily accessible in most remote clinical areas, and applicable to most patients. Moreover, it is considered to be highly reliable and preferable in the diagnosis of plantar fasciitis, diabetic foot and ankle infections, damaged soft tissues, localised cysts, heart and circulation disorders, and foetal abnormalities (Crofts et al., 2014; Angin et al., 2014; Szabo, 2013). US imaging is performed with a pulse generation and echo reflection technique that uses high frequency acoustic waves and their echoes. In this technique, the following steps take place (Kremkau and Forsberg, 2015), as illustrated in Figure 2.1 (b): (i) the ultrasonography device sends high-frequency (1-20 MHZ) acoustic or sound pulses into the targeted patient's area using different transducers (or probes); (ii) the acoustic waves (or pulses) generated by a specific probe penetrate the patient's area and reach edges between organs and tissues forming bouncing echoes; and (iii) these echoes are refracted back to the transducer, where they are detected, processed and analysed by the US device and displayed on the screen of the ultrasound device forming a 2-D or 3-D images of the targeted internal anatomic area.



(a) (b)

Figure 2.1: (a) Sonographic machine and its main components (Toshiba Healthcare), picture taken from radiology section, Salford Health sciences department), (b) a schematic diagram of a standard ultrasound imaging system showing the general mechanism of work of an ultrasound imaging system in defining the plantar fascia region in a patient's foot. According to the pulse-echo imaging concept, US imaging works by sending acoustic waves called pulses from a controlled transducer probe (with a defined frequency, time and scan mode) into the targeted scanned plantar fascia area, and receiving the refracted bouncing signals called echoes. The signal processing taking place in the central processing unit (CPU) can form a 2-D or 3-D gray-scale images to profile and display the targeted scanned plantar fascia region.

2.2 Acoustic speckle noise nature and source

Although US imaging offers many advantages in the diagnosis of plantar fascia (PF), it still experiences low contrast imaging due to marginal variation in acoustic impedance between different soft tissues ($\sim 1\%$) (Kang et al., 2016). US imaging results are largely dependent on different factors including: the operator acquiring the images, the anatomical substructure of the soft tissues, the mechanism of US systems, the transducers and frequencies used (Fornage, 1993). In addition, the quality of images can also be affected by the presence of an inherent attribute called ”acoustic speckle” that occurs in all coherent imaging modalities, including ultrasonography imaging. This is a type of multiplicative noise, which occurs during the process of US image acquisition (Ganzalez and Woods, 2002). It involves the appearance of condensed granular (bright and dark) dots in gray-scale US images when two or more reflected and scattered acoustic waves called echoes (resulting from the interaction of the travelling US pulses within the organ tissues) interfering with one another, constructively (identical echoes with a delay in arrival times and nearly cancelled amplitude) or destructively (identical echoes with nearly no delay in arrival times and nearly doubled amplitude) (see Figure 2.2) (Andria et al., 2013; Burckhardt, 1978; Goodman, 1976).

A review of speckle noise origin, its statistic properties and effects along with several speckle reduction examples can be found in (Goodman, 1976; Wagner et al., 1983; Foster et al., 1983; Szabo, 2013; Burckhardt, 1978).

The presence of speckle noise in PF US images is very common and is considered as an undesirable feature since it reduces image contrast, destroys or diffuses the image edges and affects the delineation of PF. It also affects the detection of low contrast objects that contain tiny lesions, making medical interpretation and biometric measurements challenging, and therefore impacting the accuracy of diagnosis. Furthermore, the effect of speckle noise may alter the performance of some post-processing applications such as edge detection, feature extraction, feature selection, automated segmentation and image registration. According to US and Synthetic Aperture Radar (SAR) imaging experiments performed by Wagner et al. (1983), speckle noise can be generalised as multiplicative noise (Jain, 1989) using the following equation:

$$I(i,j) = OI(i,j) * S(i,j) + \xi(i,j), \tag{2.1}$$

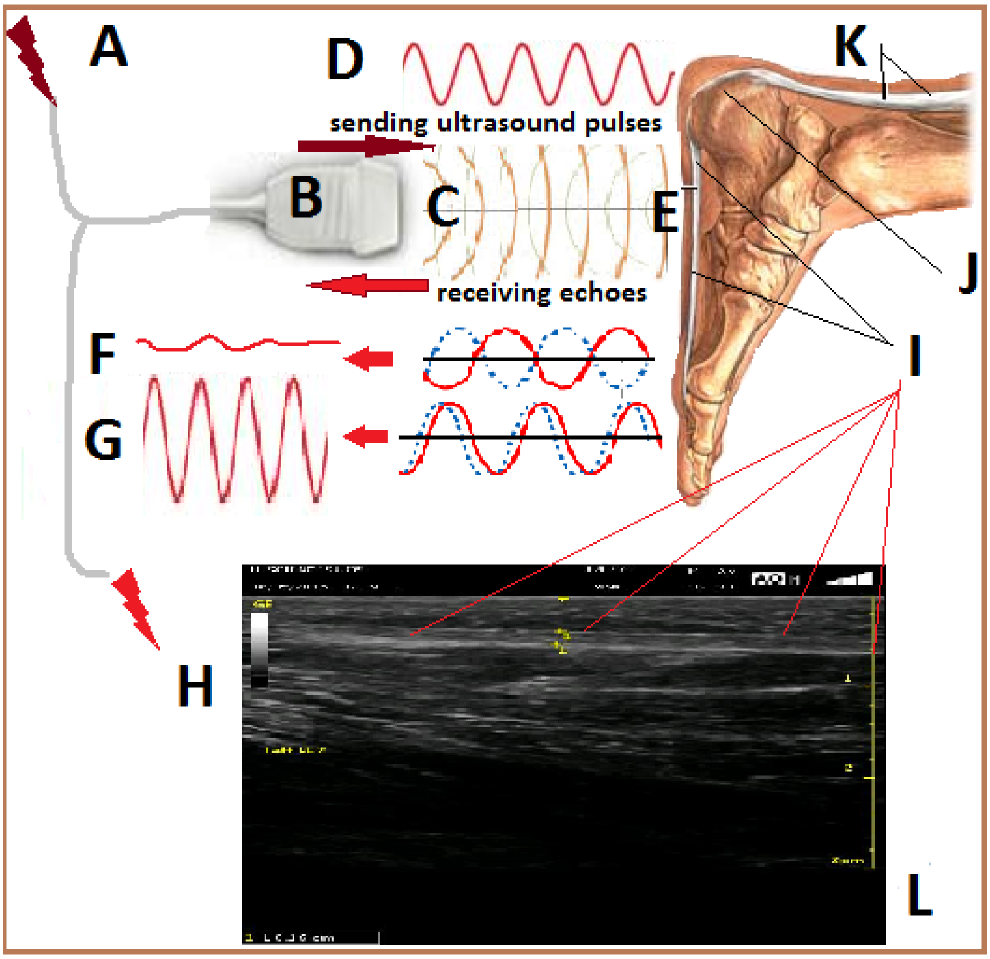


Figure 2.2: Pulse-Echo technique and acoustic speckle phenomenon source in sonographic imaging . A, Electric energy. B, Transducer probe. C, Electric energy converted to acoustic (sound) waves forming ultrasound pulses and transmitted to the targeted PF area, these pulses travel within the targeted region and interact with different tissues. D, Pulses of the ultrasound waves produced by the ultrasound transducer. E, The ultrasound waves are reflected by tissues forming echoes and returned back to the transducer in order to be processed by the CPU. F-G, Two different returning interference echoes causing the appearance of acoustic noise called speckle in ultrasound imaging: F, Constructive interference (two identical echoes with delay in arrival times and nearly cancelled amplitude). G, destructive interference (two identical echoes with almost no delay in arrival times and nearly doubled amplitude). H, Returned (reflected) acoustic waves (echoes) are converted into electrical signals and then processed by the CPU to form a 2-D or 3-D gray-scale image and display the targeted scanned area. I, Plantar fascia. J, Calcaneus. K, Achilles tendon. L, 2-D gray-scale ultrasound image showing the speckled targeted scanned plantar fascia area (I).

where $I(i,j)$ is the speckled image, $OI(i,j)$ is the original image, $S(i,j)$ represents the multiplicative part, $\xi(i,j)$ is the additive part of the speckle noise, and i,j denote the image indexes. In practice, the effect of the additive part $\xi(i,j)$ on a US image is considerably less significant than that of the multiplicative part, and it can be disregarded. Therefore, equation (2.1) can be simplified as

$$I(i,j) \approx OI(i,j) * S(i,j). \tag{2.2}$$

According to the method proposed by Jain (1989), speckle noise can be converted into additive noise by using the following equation (logarithmic transformation):

$$Log(I(i,j)) = log(OI(i,j) * S(i,j)), \tag{2.3}$$

$$Log(I(i,j)) = log(OI(i,j)) + log(S(i,j)). \tag{2.4}$$

In the homomorphic transformation, US images are logarithmically transformed, and the speckle noise is treated as an additive one. The theoretical details and the mathematical background of the despeckle filtering techniques used to suppress speckle phenomenon are presented in Sub-section 2.4.

2.3 Basic high level medical ultrasound image processing model

From ultrasound image acquisition and reconstruction to image analysis, interpretation and knowledge extraction a basic high level medical ultrasound image processing system can be introduced as a three step model: image acquisition, image processing and image understanding (Dhawan, 2011; Gonzalez and Woods, 2011) (Figure 2.3). This includes the ability to acquire high quality images (in terms of high resolution, high signal to noise ratio (SNR), and the ability to show low contrast objects that contain small lesions) using high specification imaging tools, with the ability to enhance, process, examine, extract useful diagnostic information, and make use of them in medical imaging research and applications (Dhawan, 2011).

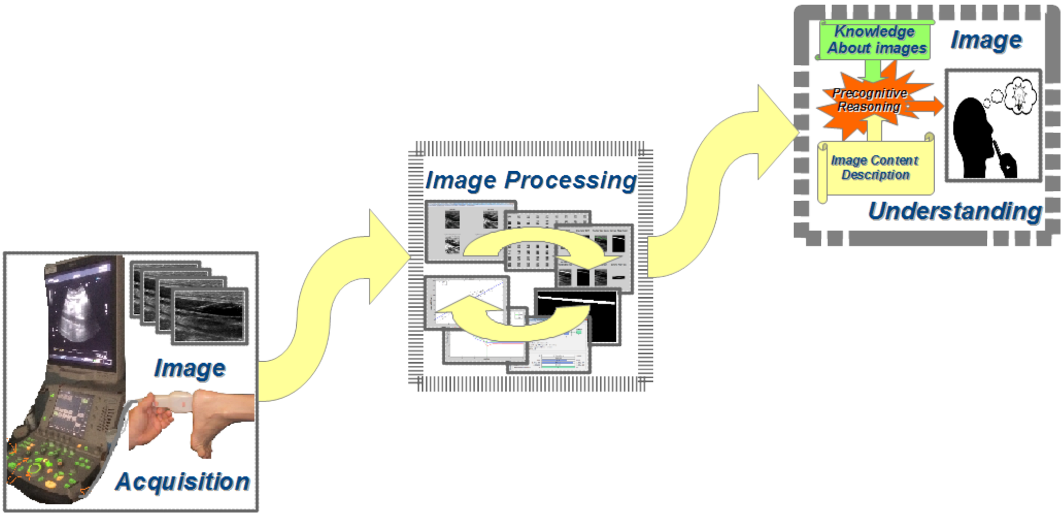


Figure 2.3: The three-step high-level medical image processing model: Image acquisition, image processing, and image understanding.

2.3.1 Ultrasound image acquisition

Ultrasound image acquisition plays an important part in medical ultrasound imaging and image formation, as the US images acquired might not have the expected results and can be defective, making medical interpretation and biometric measurements more difficult during diagnosis. Many factors affect the results of ultrasound data collection (Rueda et al., 2014). This includes: patient characteristics; ultrasound scanning machines and techniques; the type, position and orientation of the probe; image transmission and image compression; the expertise of the clinician acquiring the images; and the nature of the underlying structure of different soft tissues and organs. Since ultrasound image quality is affected by these factors, the ultrasound image processing techniques must be designed in an effective and intelligent way to improve the presentation of medical diagnostic details in the image and extract useful information (extract knowledge). In addition, the image scans must be performed according to the general medical ultrasound scan protocols as discussed in (Crofts et al., 2014), in order to generate the best possible ultrasound images (Dhawan, 2011).

2.3.2 Ultrasound image processing and analysis

Medical ultrasound image processing and analysis (some times called computer-assisted analysis) task has a great impact in the medical domain, particularly in clinical research studies and non-invasive therapy where there is no introduction of devices into the human body. It aims to support medical experts (clinicians, radiologists, physicians, etc.) in decision making during diagnosis. It is a wide and crucial subject for image analysis, modelling and content interpretation. Within ultrasound image processing, there are various elements but most of them fall into the following four main classes (Figure 2.4): (1) image preprocessing including image restoration, filtering and enhancement (for improving image visibility and its usefulness) after the effect of acoustic speckle phenomenon during image acquisition and transmission; (2) image segmentation involving the delineation of the region of interest (ROI) using various segmentation techniques to examine and analyse low contrast patterns that contain small lesions (i.e. shape analysis); (3) extraction of features and information from ROIs that could be used in classification and novelty detection processes; (4) pattern classification, novelty detection, quantitative and qualitative image interpretation for diagnosis and treatment monitoring (Umbaugh, 2010; Toennies, 2012; Dhawan, 2011) .

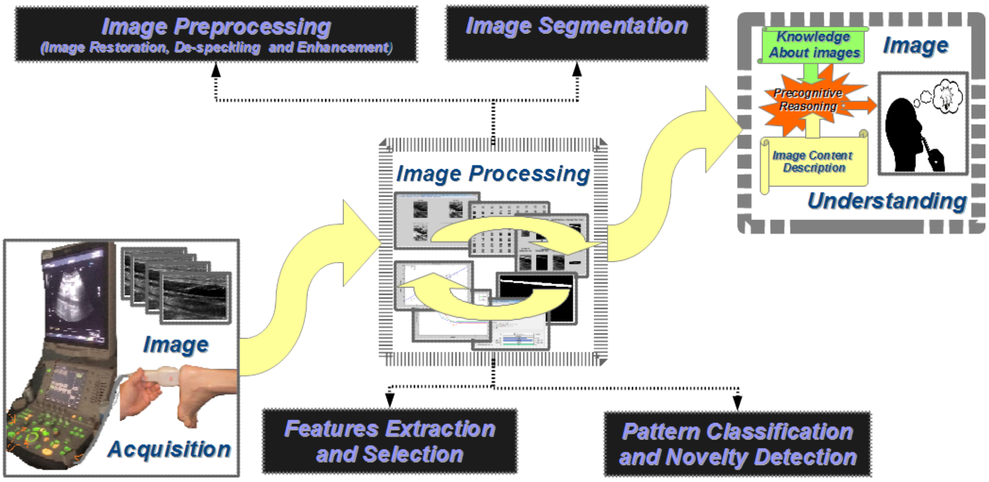


Image Preprocessing
(Image Restoration, De-speckling and Enhancement)
Image Segmentation
Knowledge About images
Image
Precognitive Reasoning
Image Content Description
Understanding
Image Processing
Image
Acquisition
Features Extraction and Selection
Pattern Classification and Novelty Detection

Figure 2.4: The main classes within image processing: Preprocessing, segmentation, features extraction and selection, pattern classification and novelty detection.

2.3.3 Ultrasound image understanding and interpretation

Image understanding can be seen as a link between human observer, computer vision (a combination of image acquisition, processing, analysis, and understanding methods). and artificial intelligence (AI), which requires reciprocal interactions of different processing stages. It is also considered as one of the most challenging problems of AI areas (such as knowledge representation, problem solving, learning from experience, semantic networks etc.) and it is always performed by physicians using knowledge based reasoning techniques (non data-driven task). The image understanding process is considered as the highest processing level of image processing model intended to figure out the relationship between the acquired images and the past established models of the real world (i.e making the model more general and widely applicable in order to match reality) (Tadeusiewicz and Ogiela, 2006; Sonka et al., 2014). This may include for example, finding out what is really happening in the acquired images, what the delineated objects are, what relationships exist between the objects and their features, and what decision should be made to match the observed world for future use and analysis.

2.4 Speckle noise filtering techniques

Analysing medical US images is difficult due to the presence of the speckle phenomenon which affects the prediction and the extraction of useful information from the US images. Many filtering approaches have been proposed in the literature to reduce speckle phenomenon without blurring or diffusing the anatomical structures of the acquired images. These approaches can be divided into three main categories: resolution improvement approaches, averaging approaches, and post-processing approaches (Lai and Dewi, 2015; Milkowski et al., 2009).

2.4.1 Resolution improvement techniques

The resolution improvement techniques focus mainly on increasing and improving the resolution of US systems in order to reduce the speckle noise using different approaches such as matrix-array transducers and higher frequency operations. However, the problem with this approach is the requirement of high operating frequency resolution resulting in large attenuation, and therefore it is not practically useful.

2.4.2 Averaging techniques

The averaging techniques are based on altering US images data and system parameters during the acquisition process by creating a single focused image from a set of multiple decorrelated image frames. Temporal averaging, spatial compounding and frequency compounding are three common averaging approaches used to reduce or eliminate speckle noise in US images, which are very expensive (Lai and Dewi, 2015; Milkowski et al., 2009).

2.4.3 Post-processing techniques

The post-processing techniques have been shown to be useful in some cases for suppressing undesirable speckle noise and improving the image quality in most medical US images. Over the past 20 years, many post-processing filters have been designed to reduce speckle noise (Finn et al., 2011; Zhang et al., 2015; Loizou et al., 2014a; Loizou and Pattichis, 2008). These filters can be broadly categorized into seven main groups: (1) non-adaptive filters, such as Mean (Jain, 1989) and Median (Loupas et al., 1989) filters; (2) adaptive local statistics filters such as mean variance (MeanV), Wiener (Jain, 1989), Lee (Lee, 1980), Kuan (Kuan et al., 1987) and Frost (Frost et al., 1982); (3) Homogeneity filtering (Homg) (Christodoulou et al., 2002; Loizou et al., 2005); (4) Geometric filtering (Geom) (Busse et al., 1995; Finn et al., 2011; Loizou et al., 2005); (5) non-linear anisotropic diffusion filters including Perona and Malik anisotropic diffusion (PMAD) (Perona and Malik, 1990), speckle-reducing anisotropic diffusion (SRAD) filtering (Yu and Acton, 2002), detail preserving anisotropic diffusion (DPAD) (Aja-Fernández and Alberola-López, 2006a), non-linear coherent diffusion (NCD) (Abd-Elmoniem et al., 2002), and oriented speckle reducing anisotropic diffusion (OSRAD) (Krissian et al., 2007); (6) wavelet transform despeckling filters, including standard discrete wavelet transform (DWT), dual tree complex wavelet transform (DT-RWT) (Rabbani et al., 2008; Michailovich and Tannenbaum, 2006), homomorphic wavelet thresholding technique (Gupta et al., 2005a), generalised likelihood method (GLM) (Pizurica et al., 2003a), spatial adaptive Wiener wavelet filtering (Shui, 2005a), and non-linear multi-scale wavelet diffusion (NMWD) filter (Yue et al., 2006c), integrating wavelet transforms with anisotropic diffusion; and (7) Hybrid model such as dual tree complex Wiener wavelet transform and HybridMedian filters.

Adaptive local filters such as Frost, Lee, and Kuan were first used by the SAR researcher community to suppress speckle noise in coherent imaging and SAR systems (Lee, 1981b). These filters have also been largely used in US imaging since the early 1980s to reduce the speckle phenomenon (Insana et al., 1989; Rabbani et al., 2008). However, several of these filters can partially reduce speckle noise and fail to retain some useful information such as high-frequency details in US images, and as a result, they cause image texture blurring and edge distortion (Liu et al., 2011). In addition, non-adaptive filters including Mean and Median were also shown to be poor in removing speckle noise from medical US images since they blur the edges and fail to preserve important diagnostic information. Due to the aforementioned issues, much research effort has been devoted to developing suitable speckle noise reduction filters in terms of preserving image edges and useful features as well as the effectiveness of the denoising filters. These include non-linear anisotropic diffusion filters, multi-scale wavelet-based filtering methods, and hybrid filters (Rabbani et al., 2008; Michailovich and Tannenbaum, 2006; Gupta et al., 2005a; Zong et al., 1998; Singh and Parui, 2006).

Recently, non-linear anisotropic diffusion and wavelet transform speckle reduction techniques have attracted considerable research attention because they are powerful tools for retrieving signals from speckled data and thus preserving edges and enhancing image contents. Recent studies using hybrid models (Yue et al., 2006a; Pizurica et al., 2003b; Portilla et al., 2003; Solbo and Eltoft, 2008; Finn et al., 2011) have shown that the integration of DWT and anisotropic diffusion can facilitate better speckle reduction and edge preservation as compared to stand-alone wavelet transform filtering and non-linear anisotropic diffusion filtering. However, such integrated approaches require further improvement. A thorough review of all these despeckling methods is beyond the scope of this thesis. For a wide range of denoising and de-speckling techniques, a recent quantitative comparative study of 48 filters can be found in (Biradar et al., 2015).

The following subsections provide an overview of the seven main groups of filtering methods and some common existing speckle reducing techniques along with their mathematical background, advantages and drawbacks.

2.4.4 Median filtering

The Median filter (Loupas et al., 1989) is considered as a spatial non-linear filter, designed for impulse (salt & pepper) and spike noise reduction (Jain, 1989) and (Ganzalez and Woods, 2002). It has been widely applied in medical imaging (Ritenour et al., 1984; Ioannidis et al., 1984), because its simplicity. Its working principle has been described by the substitution of the middle pixel in the kernel window of size $2k+1$ (k goes from 1 to N) with median value of its neighbors (Ganzalez and Woods, 2002).

2.4.5 Adaptive local statistics filtering

It is well known in the literature that most speckle reduction filtering methods use local statistics. Such filters calculate the mean weights using sub-area statistics to define the statistical measures through different Kernel windows (Loizou et al., 2005) on the assumption that the noise is a multiplicative noise as given in equation (2.2) (Lee, 1980; Loizou et al., 2002; Kuan et al., 1987; Frost et al., 1982; Loupas et al., 1989; Christodoulou et al., 2002; Jain, 1989). The kernel window size varies from 3×3 to 15×15, for this study the selected kernel window size was set to 5×5).

In this study, only two adaptive local statistics filters (MeanV and Wiener) have been selected as using other filters is time consuming and of limited practical use, including the standard Lee, Kuan and Frost filters. The main concern with this group is that they are mainly affected by the structure and the dimension of the kernel window (i.e. over-smoothing and image blurring may occur for larger kernel selection and ineffective despeckling may occur for smaller ones) (Loizou et al., 2002). The algorithms of these filters may be referred back to the same filtering approach as follows but with different weighting:

$$\begin{aligned} I_{i,j} &= \bar{K_w} + W_f \times (C_p - \bar{K_w}), Or \\ I_{i,j} &= W_f \times C_p + \bar{K_w} \times (1 - W), \end{aligned} \tag{2.5}$$

where I_{ij} is the despeckled pixel image, $\bar{K_w}$ is the mean intensity of the kernel K_w, W_f is the weighting function (represented differently in the following sub-sections according to the selected filter), C_p is the central pixel (noisy pixel value in the moving window). If the smoothing stops, the filter will output only the mean of the gray level intensity value $\bar{K_w}$.

2.4.5.1 Mean Variance

The Mean variance (MeanV) (Jain, 1989; Loizou et al., 2005) filter uses the first-order statistics (mean and variance) of every single pixel neighborhood (Suri, 2008; Loizou et al., 2005). The weighting of the mean variance can be calculated using the following equation (Hiremath et al., 2011):

$$W_f = \frac{\left(1 - \bar{K_w}^2 \sigma_k^2\right)}{\sigma_k^2 \left(1 + \sigma_n^2\right)}, \tag{2.6}$$

where σ_k^2 and σ_n^2 denote the variances in the moving kernel and the speckle noise of the whole image I respectively (the lower variance, the clearer image), they are calculated using equations (2.7) and (2.8) respectively:

$$\sigma_k^2 = \sum_{i=1}^{m} \left(\frac{\sigma_m^2}{\bar{K_m}}\right), \tag{2.7}$$

$$\sigma_n^2 = \frac{1}{M^2} \sum_{i,j=0}^{M-1} \left(I_{ij} - \bar{I}\right) \tag{2.8}$$

where σ_m^2 and K_m are the variance and mean of speckle noise in the chosen windows, respectively, m is the index of all windows in the image, $M \times M$ is the size of the image $I_{i,j}$ and $\bar{I}$ is its mean intensity (Loizou et al., 2005).

2.4.5.2 Wiener

Wiener filter (Jain, 1989) was the first approach to suppress speckle noise (Abbott and Thurstone, 1979). Wiener filtering is based on local image variance calculation (small variance value reflects a good image smoothing). (Sivakumar et al., 2010; Ganzalez and Woods, 2002). The Wiener filter is calculated using equation (2.5) with the following weighting estimator:

$$W_f = \frac{\left(\sigma_k^2 - \sigma_n^2\right)}{\left(\sigma_k^2\right)}, \tag{2.9}$$

where σ_k^2 and σ_n^2 are defined previously using equations (2.7) and (2.8) respectively.

2.4.6 Homogeneity filtering

The homogeneity filter (Homog) is mainly based on defining the neghborhood homogeneity (homogeneous area) around image pixels in order to enhance (despeckle) and preserve edges as well as flat image surfaces (Christodoulou et al., 2002). It considers exclusively pixels that belong only to the treated homogeneous neighborhood kernel area by employing equation (2.11) (Christodoulou et al., 2002; Ali and Burge, 1988; Loizou et al., 2005). The homogeneity filter can be defined as follows:

$$Y_{i,j} = \frac{\left(h_{i,j} C_{i,j}\right)}{\sum_{i,j} h_{i,j}}, \quad \text{with} \tag{2.10}$$

$$h_{i,j} = \begin{cases} 1 & if \left(1 - 2\sigma_n\right)\bar{K} \leq C_{i,j} \leq \left(1 + 2\sigma_n\right)\bar{K} \\ 0 & otherwise \end{cases} \tag{2.11}$$

where $C_{i,j}$ denotes the speckeled pixels in the kernel, $h_{i,j}$ represents the local homogeneity at a pixel (i, j), and $h_{i,j} = \sigma_s^2/\bar{K}_s$, where σ_s^2 and $\bar{K}_s$ represent the variance and mean of the moving kernel window, respectively.

2.4.7 Geometric filtering

The geometric filter (Busse et al., 1995; Crimmins, 1985), is derived from geometric concepts. It uses a non-linear speckle noise reduction technique. This technique is derived from a non-linear repeated algorithms that increase or decrease pixel's values nearby pixels neighbourhood in relation to their corresponding values (Loizou et al., 2005, 2002). This technique is derived from a non-linear repeated algorithms that increase or decrease central pixel values nearby its surroundings in relation to their corresponding values. Additinally, every kernel window central pixel is checked with the two adjacent pixels of its 8 neighbours of the following Figure 2.5 (Finn et al., 2011; Loizou et al., 2005).

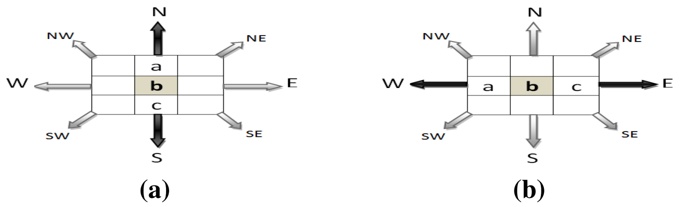


Figure 2.5: Geometric approach diagram with different pixel directions e.g.: (a) North-South selected direction and (b) West-East selected direction

The operation of the geometric filter can be summarized by the following steps as mentioned in (Finn et al., 2011; Loizou et al., 2005; Christodoulou et al., 2002; Loizou et al., 2002): 1) Select the N-S direction and assign the three pixel values ("a", "b" and "c") as in Figure 2.5(a). 2) Adjust the central pixel "b" using the following intensity adjustment rules as in (Loizou et al., 2005). 3) Repeat the previous steps (1 and 2) for the remaining directions. 4) Repeat all the previous steps (1, 2, and 3) for all pixels until the desired level of filtering is achieved.

2.4.8 Non-linear anisotropic diffusion filtering

Non-linear anisotropic diffusion (AD) is a widely used method in image recovery and image restoration (Torkamani-Azar and Tait, 1996). It offers the advantages of deep smoothing, texture enhancement and edges preservation for images affected by non-multiplicative noise (additive noise) (Uddin et al., 2013; Yue et al., 2006a). Different non-linear AD filtering methods have been proposed in the literature to suppress the speckle noise in US images including Perona and Malik anisotropic diffusion (PMAD) (Perona and Malik, 1990), speckle reduction anisotropic diffusion (SRAD) (Yu and Acton, 2002) and detailed preserving anisotropic diffusion (DPAD) (Aja-Fernández and Alberola-López, 2006b) (as presented below).

2.4.8.1 Perona and Malik anisotropic diffusion and fourth-order partial differential equation filtering

PMAD was originally proposed by Perona and Malik (1990) as a nonlinear (spatial) second-order PDE for image edge detection and enhancement in a continuous domain (Perona and Malik, 1990). The concept underlying the use PDEs in nonlinear AD for speckle suppression is extensively explained in (Weickert, 1998). The nonlinear diffusion formula (Uddin et al., 2013) can be expressed as:

$$\begin{cases} \frac{\partial}{\partial t} I\left(i,j,t\right) & = div\left[c\left(I\left(i,j,t\right)\right).\nabla I\left(i,j,t\right)\right] \\ I\left(i,j,0\right) & = I_0\left(i,j\right) \end{cases} \tag{2.12}$$

where div is the divergence factor, $\nabla I(i,j,t)$ denotes the gradient magnitude of image I, which serves as an initial step (discontinuity) for edge and boundary detection, $I_0(i,j)$ is the initial image (Uddin et al., 2013), and $c(i,j,t)$ is the diffusion coefficient represented by the following two diffusivity functions:

$$c\left(i,j,t\right) = \frac{1}{1+\left(\frac{\|\nabla(i,j,t)\|}{k}\right)^2}, \tag{2.13}$$

$$c\left(i,j,t\right) = \exp\left[-\left(\frac{\|\nabla I\left(i,j,t\right)\|}{k}\right)^2\right], \tag{2.14}$$

where k is the edge magnitude threshold (diffusion process threshold) and $\|.\|$ is the norm. In general, selecting a large value of k results in better smoothing in the homogenous area (Yue et al., 2006b). Although the popularity of PMAD technique is well documented, this method suffers from the following defects:a) blocky effects (visually unpleasant effects and detection of false edges) in images; b) loss of structural and spatial neighbourhood information; c) slow convergence; and d) poor performance in the presence of multiplicative speckle noise (although it performs well in the presence of additive noise) (Yu and Acton, 2002; Yue et al., 2006a; Pitas and Venetsanopoulos, 1990).

One proposed solution for improving the PMAD filter and eliminating blocky effects in an image is the use of non-linear fourth-order PDEs instead of second-order PDEs (You and Kaveh, 2000; Chan et al., 2000). This approach is capable of smoothing areas having small gradients, undiffusing areas having large gradients (edges and noise, if any), and avoiding blocky effects (Rajan et al., 2008). Moreover, fourth-order diffusion is considered to be much faster than second-order diffusion, and it is capable of creating a richer set of functional behaviours that can be used during image enhancement (Greer and Bertozzi, 2004). A fourth-order PDE applies Laplacian method (L^2- curvature gradient flow) (You and Kaveh, 2000), and it is given by:

$$\frac{\partial y}{\partial x} = -\nabla^2 \left[c\left(\nabla^2 I\right)\nabla^2 I\right], \tag{2.15}$$

where ∇^2 is the Laplacian transform of the image I (used for sharpening and edge detection) and $c\,(.)$ is the desirable diffusion coefficient. The different diffusivity functions used in (2.15) are described in (Yu and Wang, 2007). This study employs PeronaMalik diffusivity functions given by (2.13) and (2.14). The energy function of (2.15) is given by:

$$E\,(I) = \int_\Omega f\left(\left|\nabla^2\right|\right)\partial x \partial y, \tag{2.16}$$

where Ω is the image support and ∇^2 is the Laplacian operator. Since $f\left(\left|\nabla^2\right|\right)$ is an increasing function of $\left|\nabla^2\right|$, its global minimum occurs at $\left|\nabla^2\right| = 0$. Consequently, the global minimum of $E\,(I)$ occurs when (2.16) is satisfied. Based on (2.16), the image I is smoothed until it becomes a planar image.

$$\left|\nabla^2 I\right| = 0, \quad \forall\,(x,y) \in \Omega. \tag{2.17}$$

In order to overcome the aforementioned issues for the purpose of improving the PMAD filter, in this work non-linear fourth-order PDEs were used instead of second-order PDEs (You and Kaveh, 2000; Chan et al., 2000).

2.4.8.2 Speckle-reducing anisotropic diffusion filtering

Speckle-reducing anisotropic diffusion (SRAD) filtering (Yu and Acton, 2002) is a PDE despeckling method; it is also known as edge-sensitive diffusion method. This method outperforms the traditional Perona-Malik nonlinear diffusion (Perona and Malik, 1990), and it has been adopted for speckle reduction in SAR systems and US images because it offers the following advantages: mean preservation, variance reduction, and edge localization (Yu and Wang, 2007). Moreover, it can improve image segmentation while preserving and enhancing edges. Given an image intensity $I(x,y;t)$ estimated at positions x,y, and diffusion time index t, the SRAD function can be expressed on the basis of the continuous form of a PDE (Long and Cat, 2009) as:

$$\begin{cases} \frac{\partial I(\mathrm{x,y;t})}{\partial t} = \mathrm{div}\left[c\left(q\right)\,\nabla I\left(\mathrm{x,y;t}\right)\right] \\ I\left(x,y;0\right) = I_0\left(x,y;0\right),\left(\frac{\partial I(x,y;t)}{\overrightarrow{n}}\right)\,|_{\partial\Omega=0} \end{cases} \tag{2.18}$$

where $I(x,y;t)$ denotes the image intensity computed at location x,y, and at diffusion time t, $I_0(x,y)$ is the initial image intensity at $t=0$, div represents the divergence operator, $\partial\Omega$ denotes the edge of Ω, $\overrightarrow{n}$ represents the outside normal to $\partial\Omega$, and $c(q)$ is the diffusion factor of SRAD (Long and Cat, 2009), which can be computed as follows:

$$c\left(q\right) = \frac{1}{1+\frac{\left[q^2(x,y;t)-q_0^2(t)\right]}{\left[1+q_0^2(t)\right]}}, \tag{2.19}$$

where $q(x,y;t)$ is the instantaneous coefficient of variation (ICOV), proposed by Yu and Acton (2002) as an edge detection factor (Lee et al., 2013) and is calculated as:

$$q\left(x,y;t\right) = \sqrt{\frac{\frac{1}{2}\left(\frac{|\nabla I|}{I}\right)^2 - \frac{1}{4^2}\left(\frac{|\nabla^2 I|}{I}\right)^2}{\left[1+\frac{1}{4}\left(\frac{\nabla^2 I}{I}\right)\right]^2}}, \tag{2.20}$$

where ∇ is the gradient factor and $|.|$ denotes the magnitude. The coefficient $q_0(t)$ represents the threshold value of the diffusion function that is calculated from the homogeneous area of the image (Yoo and Nishimura, 2009) as follows:

$$q_o\left(t\right) = \frac{\sqrt{var\left[z\left(t\right)\right]}}{z\left(t\right)}, \tag{2.21}$$

where $var[z(t)]$ denotes the variance of intensity and $z(t)$ represents the mean of the homogeneous region at time t (Yoo and Nishimura, 2009).

2.4.8.3 Detail preserving anisotropic diffusion filtering

Detail preserving anisotropic diffusion (DPAD) has been proposed by (Aja-Fernández and Alberola-López, 2006b) as an improved version of SRAD filter where equation (2.19) is replaced by the following formula obtained from Kuan rather than Lee approach:

$$c\left(q\right)=\frac{1+\frac{1}{q'^{2}(x,y;t)}}{1+\frac{1}{q_{0}^{2}(t)}} \tag{2.22}$$

where $q'\left(x,y;t\right)$ is the new proposed ICOV (for less computation complexity, (Zhang et al., 2015; Finn et al., 2011)) for larger Z^2 neighbourhoods, denoted by $\eta_{x,y}$, and ICOV is calculated as follows:

$$q'\left(x,y;t\right)=\sqrt{\frac{\frac{1}{|\eta_{x,y}|-1}\sum_{p\in\eta_{x,y}}\left(I-\bar{I}\left(x,y;t\right)\right)^{2}}{\frac{1}{|\eta_{x,y}|}\sum_{p\in\eta_{x,y}}I_{p}}}, \tag{2.23}$$

2.4.9 Wavelet based filtering

In recent years, wavelet transform techniques (for transforming an image from a spatial form to a multi-resolution [frequency] form) became of great interest to researchers since they are very powerful in restoring original signals from affected or corrupted (noisy) ones (Gupta et al., 2005b). The main advantage of image signals wavelet transform is that it produces space-frequency decomposition of image signals. It overcomes the deficiency of Fast Fourier transform (FFT) (Cooley and Tukey, 1965; Brigham et al., 1988) and Discrete Cosine transform (DCT) (Ahmed et al., 1974; Rao and Yip, 2014) that are only concerned with frequency decomposition (Shih, 2010). The image wavelet transform task involves the decomposition of the image signal into a set of orthogonal essential functions through two main operations such as scaling and translation. A common approach for noise reduction in wavelet domain is to illuminate the noisy coefficients after signal decomposition, compose and restore the free-noise image using composition filters (Dhawan, 2011). The 2D DWT, DT-RWT, and DT-CWT are some of these approaches that have been widely used in ultrasound image de-speckling applications (Rabbani et al., 2008; Michailovich and Tannenbaum, 2006). Their descriptions and implementation details can be found in (Kingsbury, 1998, 1999; Selesnick et al., 2005; Freeman and Adelson, 1991; Sendur and Selesnick, 2002a; Selesnick, 2002, 2001).

2.4.9.1 Dual tree complex wavelet transform filter

Recently, many techniques have been proposed to improve the performance of wavelet-based image filtering. Kingsbury (1998; 1999) was the first to introduce dual tree complex wavelet transform (DT-CWT) as a solution to overcome the limitations of the standard distrect wavelet transform (DWT), based on the following properties (Rizi et al., 2011; Serbes and Aydin, 2010; Selesnick et al., 2005): (a) two separate trees of real filters for creating the wavelet coefficients parts (real and imaginary parts), as shown in Figure 2.6; (b) shift invariance approximation to a high degree; (c) good directional selectivity in 2D data; and (d) efficient computation with limited redundancy.

Typically, the standard DWT produces three bandpass sub-images at each level, which correspond to horizontal, vertical, and diagonal coefficients, and are oriented at angles of $0^o, \pm 90^o$, and $\pm 45^o$. In contrast, DT-CWT can produce six sub-images at each level, and oriented at six fixed angles: $\pm 15^o, \pm 45^o$, and $\pm 75^o$ in 2D. This directional information is useful for determining the finest presentation of the image features. The complex wavelet-based function is given by:

$$\psi_c = \psi_r(t) + j\psi_i(t), \tag{2.24}$$

where $\psi_r(t)$ represents the real part of the complex wavelet, $\psi_i(t)$ denotes the imaginary part. The implementation details of the DT-CWT can be found in (Kingsbury, 1998, 1999; Selesnick et al., 2005).

2.4.9.2 DT-CWT wavelet thresholding (shrinkage)

Denoising and despeckling methods involving wavelet-based thresholding are widely used in the wavelet domain, where high coefficients are the real signal and low coefficients represent the image noise. The objective of thresholding is to eliminate all (high-frequency sub-band) coefficients that are below the coefficient threshold by setting them

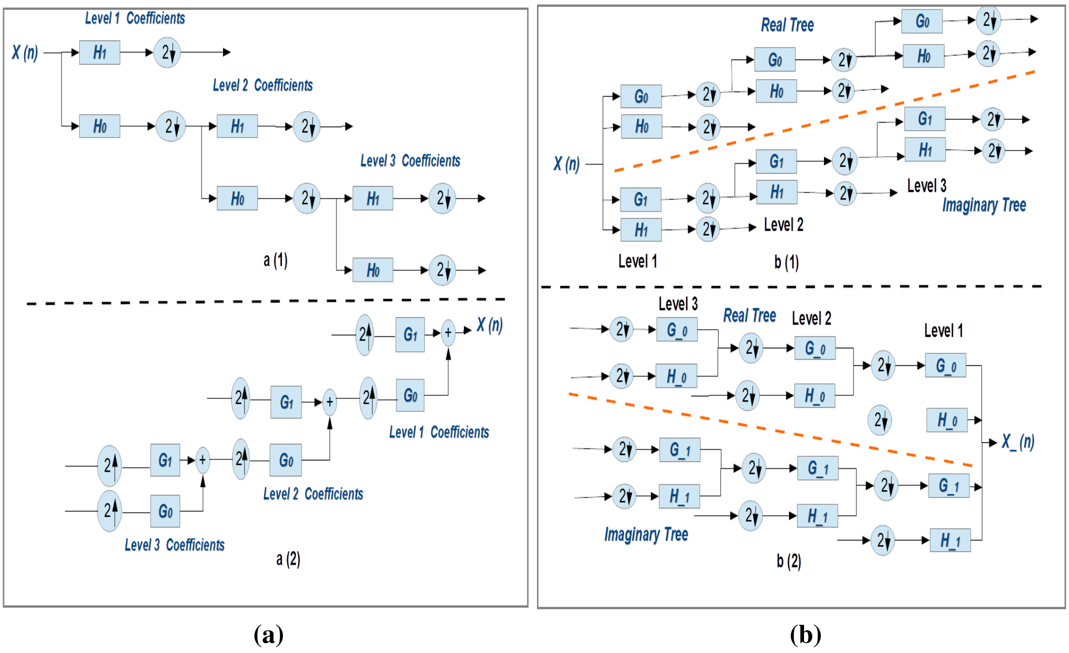


Figure 2.6: DWT vs DT-CWT - A schematic tree diagram showing a multiresolution of three-level signal decomposition and composition for both: (a) DWT, and (b) DT-CWT wavelet filters. (a) DWT: a(1) three-level signal decomposition using low pass (H_0) and high pass (H_1) decomposition in order to obtain DWT coefficients, a(2) three-level signal composition from DWT coefficients by applying low pass (G_0) and high pass (H_1) composition filters. (b) DT-CWT decomposition and composition showing the real tree and imaginary tree: b(1) three-level signal decomposition using both the real part tree (with low pass (G_0) and high pass (H_0)) decomposition and imaginary tree (with low pass (G_1) and high pass (H_1)) decomposition to obtain DT-CWT coefficients, b(2) three-level signal composition from DT-CWT wavelet coefficients using low pass (G_0) and high pass (H_0) of real composition filters and low pass (G_1) and high pass (H_1) of imaginary composition filters.

to zero (Borhani et al., 2005). Wavelet-based thresholding for despeckling US images can be considered as an estimation problem in which the true image signal component is to be recovered from the degraded image signal that is affected by the speckle noise component. This method was originally developed by Donoho and Johnstone (1995), who computed the estimation using “thresholding estimator on an orthonormal basis $B = \{g_m\}_{0 \leq m < N}$” (Jin et al., 2005) as follows:

$$\hat{X} = \sum_{m=0}^{N-1} \rho_m \left(\langle X, g_m \rangle \right) g_m, \tag{2.25}$$

where ρ_m denotes the thresholding function for eliminating the noise components (Jin et al., 2005). Donoho (1995) proposed two basic thresholding methods, namely, hard thresholding and soft thresholding. They are the most commonly used techniques for wavelet-based denoising. In this thesis, only four thresholding methods are addressed: hard, soft, trimmed, and bivariate shrinkage thresholding.

- The hard thresholding rule is defined in (Saurabh et al., 2015) as:

$$\rho_T(x) = \begin{cases} x, & \text{if } |x| > T \\ 0, & \text{if } |x| \leq T \end{cases} \tag{2.26}$$

- The soft thresholding rule is defined in (Saurabh et al., 2015) as:

$$\rho_T(x) = \begin{cases} x - T, & \text{if } |x| > T \\ x + T, & \text{if}\, x \leq -T \\ 0, & \text{if } |x| < T \end{cases} \tag{2.27}$$

where x represents the wavelet coefficient, T denotes the threshold index, and $\rho_T(x)$ is the filtered wavelet coefficients (Saurabh et al., 2015). In the hard thresholding method as defined by (2.26), wavelet coefficients that are lower than the threshold index T are cancelled (set to 0), whereas others are kept unchanged. The soft thresholding technique is considered as an updated version of hard thresholding. First, it eliminates coefficients $< T$; then, it separates the real signal from the noise in the remaining coefficients by computing the difference between them and the threshold index and setting the non-zero coefficients results to zero (Prinosil et al., 2010), as shown in (2.27). Both hard and soft thresholding are shown in Figure 2.7, and the results are compared with the original signal.

In spite of the widespread use of hard and soft thresholding in wavelet-based denoising, these techniques suffer from the following drawbacks: (1) soft thresholding faces the problem of having large bias value due to the threshold T of large wavelet coefficients; and (2) hard thresholding faces the problem of having large variance and instability due to lack of discontinuities in (2.26) (Prinosil et al., 2010; Vidakovic, 2009). Therefore, several researchers (Zang et al., 2009; Fang and Huang, 2004; Lin and Cai,

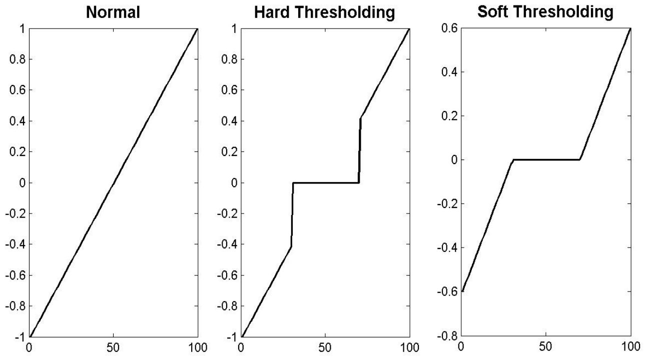


Figure 2.7: Hard and soft thresholding functions

2010; Zhang et al., 2008; Cai-lian et al., 2010) have proposed alternative thresholding methods to improve conventional thresholding approaches.

- Wavelet trimmed thresholding

Wavelet-based trimmed thresholding was been first proposed by Fang and Huang (2004) as an improved approach of hard and soft thresholding. This technique is defined by (2.28) and shown in Figure 2.8. It was suggested that careful selection of the factor α for a specific signal can give the best filtering results, as shown in Figure 2.9.

$$\rho_T(x) = \begin{cases} x\left(\frac{|x|^{\alpha}-T^{\alpha}}{|x|^{\alpha}}\right), & \text{if } |x| \geq T \\ 0, & \text{if } |x| < T \end{cases} \tag{2.28}$$

where α is a parameter for a particular signal. When $\alpha = 1$, soft thresholding is achieved, and when $\alpha \to \infty$, hard thresholding is achieved. Figure 5 shows the root mean square error calculation for different α values. Accordingly, in this work, $\alpha = 7$ has been selected with low RMSE value.

- Bivariate shrinkage thresholding

Sendur and Selesnick (2002a; 2002b) proposed a new statistical model using a bivariate probability distribution function (PDF) for filtering wavelet coefficients in the natural images (Sendur and Selesnick, 2002a,b). They used Bayesian estimation theory to derive a non-linear shrinkage function from this model for wavelet denoising. This approach has been considered as a generalization of soft thresholding approach

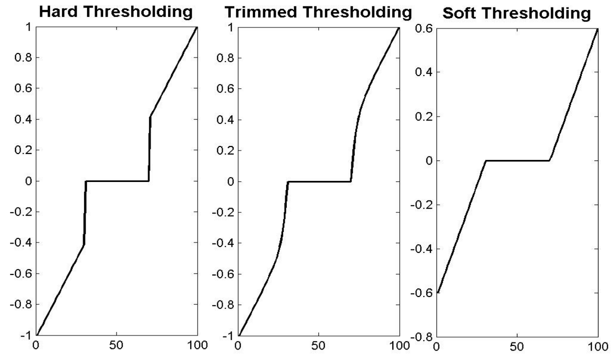


Figure 2.8: Trimmed thresholding compared to hard and soft thresholding (threshold $T = 0.4$ and $\alpha = 7$).

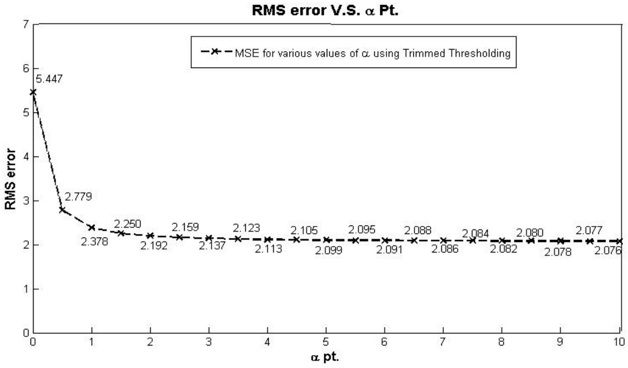


Figure 2.9: RMSE for different values of α generated using Matlab software.

of Donoho and Johnstone, and claimed to be one of the effective image filtering approaches in the literature (Chen and Qian, 2011). For any given wavelet coefficient w_i, let w_j be the parent of w_i. In general,

$$y = w + n, \tag{2.29}$$

where $y = \left(y_i, y_j\right)$, $w = \left(w_i, w_j\right)$, and $n = \left(n_i, n_j\right)$ are the noisy (wavelet) coefficients, the filtered coefficients, and the (Gaussian white) noise, respectively (Chen and Zhu, 2008). The proposed non-Gaussian bivariate PDF is given by:

$$p_w(w) = \frac{3}{2\pi\sigma^2} \exp\left(-\frac{\sqrt{3}}{\sigma}\sqrt{w_1^2 + w_2^2}\right), \tag{2.30}$$

where σ denotes variance of the signal (Lal et al., 2009). The bivariate thresholding function (plotted in Figure 2.10) is given by:

$$w_1 = y_1 . \left(1 - \frac{\frac{\sqrt{2}}{\sigma}\sigma_n^2}{\sqrt{y_1^2 + y_2^2}} \right)_+ , \tag{2.31}$$

where σ_n^2 is the noise variance. The "+" sign at the end of (2.31) denotes positive values; otherwise, it is set to zero and can be formulated as $(x)_+ = max\,(x, 0)$.

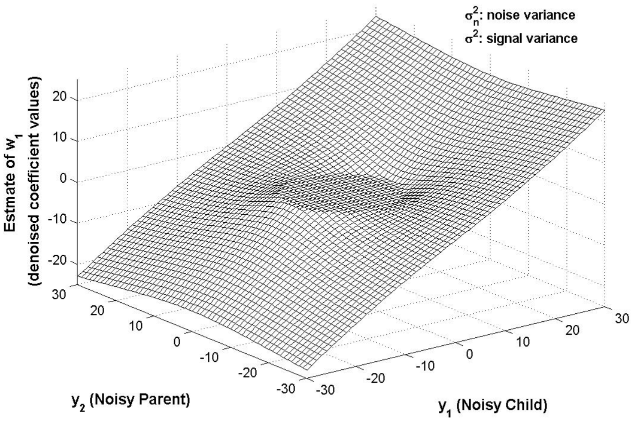


Figure 2.10: Bivariate shrinkage thresholding function.

2.4.9.3 Threshold selection and calculation rules

The selection of the threshold rule is a crucial step in wavelet-based denoising, because the performance of a wavelet-based filter depends mainly on the thresholding method and careful selection of the threshold for a given denoising application. In general, the selection of a low threshold leads to insufficient noise suppression; on the other hand, the selection of a high threshold leads to over-smoothing and distortion of useful information. Different threshold selection rules have been followed by various researchers to estimate and select the threshold value. The most commonly used threshold calculation rules in the literature are summarized below.

2.4.9.3.1 VisuShrink threshold

The VisuShrink threshold was first introduced by Donoho and Johnstone (1994; 1995). It uses the universal threshold defined by (2.32) (Hiremath et al., 2011),

$$T = \sigma\sqrt{2\log\left(N_c\right)}, \tag{2.32}$$

where N_c represents the size of wavelet coefficients, σ is the noise standard deviation and can be computed using the median absolute deviation factor (MAD) (Prinosil et al., 2010) as:

$$\sigma = \left(\frac{median\left(HH_1\left(x,y\right)\right)}{0.67452}\right), \tag{2.33}$$

where x and y are pixel indexes of HH_1, which represents the diagonal sub-band of first-level wavelet decomposition of the testing image. Chang et al. (2000) noted that VisuShrink threshold can yield overly smoothed images and reduce too many useful coefficients. This is attributable to its selection of a universal threshold, which can be very large because it is directly related to the image size value (N_c) for a typical image of size 512×512.

2.4.9.3.2 SureShrink threshold

SureShrink threshold combines the universal threshold and the threshold obtained from Stein's unbiased risk estimator (SURE) proposed by Donoho and Johnstone (1995) (Chen et al., 2007). It calculates the threshold index for each wavelet sub-band level (Nason, 1995), followed by soft thresholding. The SureShrink threshold is considered to be suitable for images with sharp discontinuities, providing good noise reduction performance and low mean square error values (Andria et al., 2013; Chang et al., 2000; Chambolle et al., 1998). The SureShrink threshold rule is given by:

$$T = min\left(t, \sigma\sqrt{2\log\left(N_c\right)}\right), \tag{2.34}$$

where t is SURE reducing value for threshold calculation and estimation, while σ and N_c are already defined previously (Om and Biswas, 2012).

2.4.9.3.3 BayesShrink threshold

The BayesShrink threshold was first proposed by Chang et al. (2000) to minimize the Bayesian risk. In practice, it is considered to be similar to the SureShrink threshold, because it is an adaptive, sub-band-dependent thresholding method that performs soft thresholding. However, it yields better results when the wavelet coefficients are modeled using general Gaussian distribution (GGD) within each sub-band.

The BayeShrink threshold for a given sub-band Ws of an image is given by:

$$T_s = \frac{\sigma_D^2}{\sigma}, \tag{2.35}$$

where T_s and s are estimated thresholds and variances in different wavelet sub-bands, respectively (Wu and Wang, 2010), and σ_D represents the computed noise variance proposed by Donoho and Johnstone using the following estimator:

$$\sigma_D^2 = \left[\frac{median\left(HH_1\left(x,y\right)\right)}{0.67452}\right]^2, \tag{2.36}$$

where 0.67452 is MAD value of normal distribution (with 0 mean and unit variance), HH_1 denotes the finest level sub-band at x and y pixel indexes, and σ is the computed signal variance in different wavelet sub-bands (Hiremath et al., 2011), and it is given by:

$$\sigma = \sqrt{max\left(\sigma_n^2 - \sigma_D^2, 0\right)}, \tag{2.37}$$

where σ_n are the calculated coefficient variances in the different sub-bands (Wu and Wang, 2010), and they can be obtained empirically as:

$$\sigma_n^2 = \frac{1}{m \times n} \sum_{i,j=1}^{m,n} W_s^2, \tag{2.38}$$

where $m \times n$ represents the sub-band coefficients size, and W_s are detail wavelet coefficients (high-frequency coefficients) in HH_1 (Wu and Wang, 2010).

In general, most existing thresholding methods and threshold selection rules rely on the assumption that images are affected by Gaussian noise (additive noise), and they lack the ability to effectively eliminate multiplicative noise (speckle) from medical US images. Therefore, it is necessary to employ homomorphic wavelet filtering, whereby logarithmic transformation is first applied to noisy US images in order to convert the multiplicative noise model into an additive model, then, the exponential operation is performed after applying wavelet transformation to the log-transformed US images.

2.4.10 Hybrid filters

2.4.10.1 Doulby local Wiener filtering with directional windows hybrid filter

Integrating adaptive local statistics filtering such as Wiener in the wavelet transform field is an efficient noise reduction hybrid technique. This integration aims to improve the image quality performance and to minimize the computational complexity risk. Integrating adaptive local wiener filtering in the wavelet based field required many steps, some of them were discussed earlier in wavelet based filtering section, among these steps two significant computation should take place (Shui, 2005b; Shui and Zhao, 2007): (1) the computation of the signal variance of each wavelet sub-band using equation

$$\sigma_n^2 = \frac{1}{m \times n} \sum_{i,j=1}^{m,n} W_s^2; \tag{2.39}$$

(2) the estimation of the signal wavelet coefficients for all sub-bands and scales using the integrated adaptive local wiener filter (and applying the wiener filter on the noisy coefficients):

$$\hat{s}_n = \frac{\sigma_n^2}{\sigma_n^2 + \sigma_\varepsilon^2} \times y_n, \tag{2.40}$$

where σ_n^2 denotes the signal variance, σ_ε^2 represents the variance and y_n is the noisy wavelet coefficients.

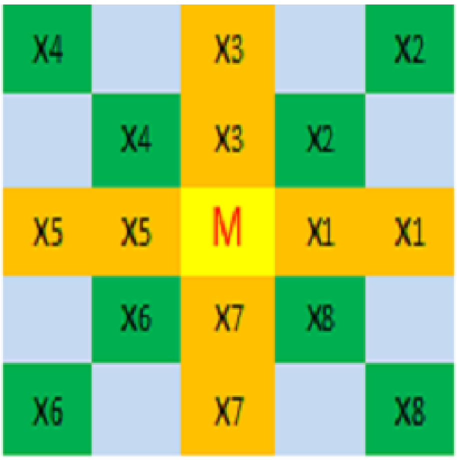
As an example of this integration, Peng-Lang Shui, (2005b) introduced a double local Wiener denoising method (DLWFDW) in the wavelet based domain. This technique employs the following features: (1) the use of elliptic orientation windows for different sub-bands (Horizontal, vertical, and diagonal windows); (2) two different groups of local Wiener filtering were applied on the noisy images using the previous three directions; (3) the use of two different wavelet based transforms approaches (2 decimated (DWT) or 2 undecimated wavelet transform (UDWT)). In the process of DLWFDW filtering, the first wiener filtering group applies elliptic orientation windows of medium size to get the first less filtered image (called pilot image). In the second wiener filtering group, the signal variance of each wavelet sub-band is computed from the output of the first group (pilot image) with the orientation windows size smaller than the first group. The full implementation of DLWFDW filter can be found in (Shui, 2005b).

2.4.10.2 Hybrid median filter

Hybrid median filter (HybridMedian) (also called corner-preserving median filter) (Russ, 2016) is an improved version of the Median filter, which was first introduced by Nieminen et al. (1987) and subsequently used in different US images despeckling applications (Loizou et al., 2014b; Loizou and Pattichis, 2008). It calculates the median of the filtering results generated by the median filter employing three different kernel window shapes such as (90^o) horizontal/vertical-shape window, (45^o) diagonal-shape window and standard shape window (Loizou et al., 2014b; Russ, 2016). In a big filtering kernel window more sub-neighbourhood directions and orientations can be introduced; for example a 5x5 kernel window (Figure 2.11) needs 4 orientations (Horizontal,vertical in yellow and two diagonals in green), giving 4 output values (according to 4 orientations) that can be ranked using three iterations of the main central pixel value. The final central-pixel M' value can be computed as:

$$M' = med(M, med(M, X1, X3, X5, X7) med(M, X2, X4, X6X8)), \tag{2.41}$$

where M-X8 represent the sub-filters in the 5x5 kernel window. M and M' are the initial central pixel and the final calculated central pixel value, respectively. Unlike the Median filter, HybridMedian improves the global image quality and preserves image edges (lines and corners) (Russ, 2016). But on the other side, it suffers from smoothing the images due to the number of iterations and it is at the risk of a computational penalty when using a large kernel window (Zhang et al., 2015; Loizou et al., 2014a; Biradar et al., 2015) .



X4		X3		X2
	X4	X3	X2	
X5	X5	M	X1	X1
	X6	X7	X8	
X6		X7		X8

Figure 2.11: Diagram illustrating the neighbourhood pixels used in 5x5 kernel window of the Hybrid-Median filter. It shows 4 orientations including the central pixel M (Horizontal,vertical in yellow and two diagonals in green), the blue pixels are being ignored. It gives 4 output values (according to 4 orientations) that can be ranked separately using three iterations as in equation (2.41).

2.4.11 Summary of speckle-noise filtering

Reducing noise from ultrasound images is a challenging process for researchers. There are several approaches and models for speckle noise reduction in the literature. However, most of these approaches have certain shortcomings that can be outlined as follows: (1) some of these filters are affected by the wrong selection of the size and shape of the kernel window (i.e. over-smoothing and image blurring may appear for large size and shape selection, and ineffective despeckling may occur for small ones); (2) some filters are based on experimental threshod estimation during the filtering process which may lead to ineffective filtering especially at the edges and boundary areas; and (3) some of the existing speckle noise reduction techniques require many iteration steps and this leads to high computational time, and others are incapable of preserving and enhancing the image edges (they are only successful near the edges). It has been noted from previous research that a good speckle noise reduction filter should focus on the following aspects: (a) speckle noise should be reduced or removed from ultrasound images; (b) image edges should not be diffused or lost and should be clear and well defined; (c) texture details and global contrast should be preserved; (d) no artefacts should be present in the de-noised ultrasound image.

Careful selection of denoising filters at this stage is very important in despeckling the plantar fascia ultrasound (PF) images. Therefore, a comparative evaluation attempt has been made to analyse the quantitative ability of the above mentioned filters based on feature extraction analysis, ANN classification, qualitative evaluation metrics (such as: MSE, RMSE, SNR, PSNR, AD, UQI, SSIM, CoC, and EPI) and visual evaluation by clinical experts in the assessment of PF US images. It is also envisaged from this study that filters presenting a superior edge preserving behaviour and best filtering results can be used as a preprocessing step for the following stages (automated segmentation, classification and novelty detection).

2.5 Image contrast enhancement

Due to the complexity of ultrasound images, various sets of tissues and anatomical structures involved, image enhancement techniques are needed for improving the visibility and detectability of the region of interest. Contrast enhancement techniques are widely used in different medical image processing applications (Acton, 2013; Pizer, 2003; Pizer et al., 1987). Among these techniques, histogram equalization (HE) (Kim, 1997; Gonzalez and Wintz, 1977), global histogram equalization (GHE) (Kim et al., 2001), local histogram equalization (LHE) also known as adaptive histogram equalization (AHE) (Kim et al., 1998) and contrast limited adaptive histogram equalization (CLAHE) (Zuiderveld, 1994). HE is very popular due to its simplicity, it improves the image contrast by reallocating the gray-scales histogram values based on the probability distribution of the input gray-scales to a new range of histogram values (it stretches the range of the image histogram) (Martínez-Trinidad et al., 2006). However, this method suffers from loss of contrast in low frequency regions and over-contrast enhancement in high frequency image regions (Kim, 1997; Lu et al., 2010). GHE is based on a global technique which applies a global contras enhancement on the whole image. But this approach fails to deal with the local brightness features. As consequences, AHE is designed to improve the GHE approach. It takes the advantage of HE features and the integration of the local statistical information of the image using a sliding window (slides through pixels). However, AHE is still experiencing some difficulties such as computational complexity; occasionally, leads to over-contrast enhancement and noises visibility enhancement (Acton, 2013; Martínez-Trinidad et al., 2006). In medical imaging, CLAHE (Zuiderveld, 1994) is advantageous in enhancement of low-contrast images when compared to standard adaptive histogram equalization and AHE approaches (Pizer et al., 1987). The CLAHE performs its function by splitting the input image into dependant sub-regions (tiles), where histogram equalization was applied on each one. The neighbouring sub-regions are combined by using a bi-linear interpolation operation to avoid artifact. This could improve the contrast and gives efficient results (Zhao et al., 2010; Lu et al., 2010).

The histogram of any digital image can be computed by the following discrete function with intensity levels in the range $[0, L-1]$,

$$H(r_k) = n_k, \tag{2.42}$$

where r_k is the $k^t h$ intensity value and n_k is the number of pixel in the image with intensity r_k. The normalized histogram is calculated using the following probability density function (PDF),

$$P_r(r_k) = \frac{n_k}{MN}, \qquad k = 0, 1, 2, ..., L-1, \tag{2.43}$$

where $P_r(r_k)$ is an estimated of the probability of occurrence of intensity level r_k in an image. $M \times N$ represents the total number of pixels in the image. The sum of all PDF components is equal to 1. The histogram equalization is obtained by next equation:

$$S_k = (L-1)\sum_{j=0}^{k} P_r(r_j), \qquad k = 0, 1, 2, ..., L-1, \tag{2.44}$$

where S_k is the new distribution of the histogram.

According to Zuideveld (1994), CLAHE approach consists of the following steps:

- the input image is split into several non-overlapping sub-regions (tiles);
- the histogram equalization is then applied on each region;
- the contrast expansion clip limit value is computed using equation (2.45) for clipping the histograms;

$$\beta = \frac{MN}{L}\left(1 + \frac{\alpha}{100}\left(S_{max} - 1\right)\right), \tag{2.45}$$

where β and α are clip limit and clip factor respectively, if $\alpha = 0$, the clip limit equal to $\left(\frac{MN}{L}\right)$, furthermore if $\alpha = 100$, the maximum allowable slope is S_{max}.

- each histogram is reallocated according to the computed clip limit values;
- gray-scale mapping of the resultant limited histograms, using cumulative distribution function (CDF). These steps are more explained in (Zuiderveld, 1994).

Owing to the standard adaptive histogram equalization and AHE aforementioned issues, in this work the CLAHE enhancement technique was selected to adjust the intensity of the PF region after the speckle noise reduction task in order to prevent speckle noises visibility enhancement (amplification) and to improve the PF delineation results.

2.6 Image segmentation techniques

Image Segmentation is considered as an important task in medical image processing and analysis for clinical evaluation and computer aided diagnosis (CAD). It is an inevitable key step for other image processing tasks such as feature extraction, detection, selection, shape analysis, pattern classification and novelty detection. It is mainly used to locate region of interest objects and boundaries in images. It is considered to be the most challenging task in medical US images over other imaging modality such as CT and MRI due to attenuation, speckle, shadows, and signal dropout. Furthermore, there is no common accepted method for US image segmentation, because segmentation techniques vary widely according to every specific problem, application, imaging modality and human interaction, and due to the homogeneity of images, spatial characteristics of the image, continuity, texture and image content (Noble and Boukerroui, 2006; Rueda et al., 2014).

There are many techniques developed for image segmentation process, they can be categorised into four main classes (Kim and Joukov, 2016): (a) thresholding, (b) edge-based, (c) region-based, and (d) special theory-based techniques (Fu and Mui, 1981; Ping, 2004; Kang et al., 2009). In another study (Pal and Pal, 1993), thresholding class has been considered as a special sub-class of region extraction technique; more details about different classification schemes of various segmentation techniques can be found in (Zhang, 2006). Full description of these techniques is presented in the following subsections.

2.6.1 Thresholding-based image segmentation technique

Thresholding-based technique is the simplest method of image segmentation related to image space regions (the characteristics of the image). This technique determines a proper threshold (grey-level intensity value T), then divides the image pixels (of the binary image) into several classes by separating the foreground objects (with intensity greater than the threshold) into one class and the background (with intensity less than the threshold) into another class (Kang et al., 2009; Pham et al., 1998). The basic thresholding procedure can be summarised as follows:

$$I_{seg}(x,y) = \begin{cases} 1, & \text{if } I(x,y) > T \\ 0, & \text{if } I(x,y) > \leq T \end{cases} \tag{2.46}$$

where $I_{seg}(x,y)$ denotes the segmented binary image with two different groups of binary grey outputs: ”1” representing foreground object regions and ”0” for the black background, $I(x,y)$ is the original gray-scale image, and T represents the gray value threshold defined at the valley peak point from the histogram representation.

There are different types of thresholding techniques (Hum, 2013): (a) global (e.g. Otsu (Otsu, 1975)) where, only one threshold value is defined for the whole image based on the histogram statistics of the image; (b) local (e.g. simple statistical thresholding, 2-D entropy-based thresholding, histogram-transformation thresholding, etc.), the threshold value is derived from the local image properties (e.g. local average grey value of the image) (Chang and Lie, 2006); (c) dynamic or adaptive (e.g. watershed and interpolatory thresholding), if the threshold values are calculated separately from each pixel or set of pixels in the image using a sliding kernel window over the input image (Kang et al., 2009; Sahoo et al., 1988). The main issues related to this technique are the choice of the threshold value that gives only two different classes (e.g. foreground and background) which cannot be used in multi-channel images (Baradez et al., 2004). Another critical problem of thresholding is the manual selection of the threshold values and the kernel window size which tends to be difficult, time consuming and computationally expensive when the number of sub-image regions increases (Hum, 2013; Buie et al., 2007). Moreover, thresholding does not consider the spatial characteristics of the image which tend to be sensitive to noise and grey level intensity inhomogeneity, and this will also corrupt the histogram of the image, making the partition very hard (Pham et al., 1998). As an improvement for the above limitations, multilevel thresholding (segmenting the image into multiple classes) and automatic threshold selection algorithms are proposed (Yan et al., 2005; Tsai, 1995). However, these improvements fail when there is limited difference in intensity distribution between the foreground objects and background due to gray level intensity overlapping (Whatmough, 1991; Buie et al., 2007) (as in our case when segmenting the PF ultrasound images using thresholding methods). A survey of the majority thresholding methods is presented in (Sahoo et al., 1988; Hum, 2013).

2.6.2 Edge-based techniques

Edge detection approaches are very common and widely used techniques in medical image processing and object delineation. This technique is based on detecting edges or the boundaries between two or more different areas in an image based on the grey-scale properties and discontinuities. These techniques are usually used as a preprocessing of another segmentation approach. In general, there are two main techniques used here, gradient-based methods and grey histogram (zero-crossing) methods (Kang et al., 2009; Umbaugh, 2010). However, these methods are unsuccessful when applied to images that are edgeless and very noisy (Umbaugh, 2010) (especially PF ultrasound images affected by speckle noise, where there is not much difference between foreground and background). As a partial solution to this, a post-processing stage of edges and boundaries tracking, linking or grouping is proposed (e.g. using Hough transform (Hough, 1962) as a linking and line detection algorithm). But this solution is considered computationally intensive and not very efficient (Dhawan, 2011; Sridevi and Sundaresan, 2013). The most traditional algorithms used in image edge detection are Sobel (Duda et al., 1973), Prewitt (Prewitt, 1970), Laplacian (Reuter et al., 2009), and Canny (Gonzalez and Woods, 2011) operators.

2.6.3 Region-based techniques

Region based segmentation techniques split images into segments that are similar based on a set of predefined similarity criteria such as intensity and edges information (Kang et al., 2009, 2012). There are two common techniques used in this approach; region growing and region splitting (Shi and Malik, 2000; Umbaugh, 2010; Kang et al., 2012; Dhawan, 2011).

2.6.3.1 Region-growing techniques

A Region-growing technique is considered as a pixel-based segmentation method, which first selects initial pixel points usually called seeds and selected manually representing well-defined image regions and then grow them to classify the all of the image pixels into sub or larger regions based on some predefined rules. In particular, these rules define the growth mechanism and examine areas homogeneity after each growth stage, e.g. checking grey level histogram values, color and edges in the image (Chang and Li, 1994).

2.6.3.2 Region-splitting and merging techniques

Region-splitting and merging technique is the opposite concept of region-growing method. It first splits the whole image which is considered as a homogeneous into a set of arbitrary sub-image objects without choosing the initial seed points until all image regions become homogeneous and then merges these regions according to their similarity features (e.g. pixels' gray level intensity values). (Kang et al., 2009; Pal and Pal, 1993).

The main issues of this kind of segmentation approaches are: (a) they are computationally expensive and time consuming as they are based on iterative algorithmic operations; (b) some of these techniques require a manual interaction to obtain the seed points; (c) they are sensitive to image noise; and (d) it can be hard to define and adjust the homogeneity and similarity rules because failure to do so will affect the segmentation results (e.g. over or under segmentation and fragmentation may occur) (Pham et al., 1998; Kang et al., 2009; Pal and Pal, 1993; Chang and Li, 1994; Shi and Malik, 2000; Umbaugh, 2010). As a partial solution to these shortcomings, homotopic region growing to protect the topology between initial and selected regions (Mangin et al., 1995) and fuzzy analogies to region-growing (Udupa and Samarasekera, 1996) algorithms have been developed. In the context of PF region segmentation, region-based segmentation fails to segment PF regions that lack definition in their homogeneity and similarity rules, where there is not much difference between PF region and the surrounding tissues.

2.6.4 Theory- or model-based techniques

Automatic medical ultrasound image segmentation is a challenging task because it deals with very low quality, and noisy images to locate the region of interest and its boundaries. Generally, the lack of contrast and the existence of speckle noise and artifacts will lead to false delineation and missing edges and boundaries in the segmented region of interest. Theory-based techniques are automatic segmentation approaches derived from other domains of knowledge (e.g. pattern recognition and mathematics fields). They aim to overcome the previously discussed challenges and to translate the medical experts' knowledge about the objects (e.g. shape, delineation, appearance, anatomical structure and exact location in the image) into intelligent computer-based algorithms that posses a priori information about the anatomical structure of the region of interest (Deserno, 2011). Several previous works on automated theory-based segmentation have been carried out on different ultrasound images to extract a variety of structures such as ovarian cysts (Zimmer et al., 1996), echocardiograms (Sebbahi et al., 1997), the calcaneus in broadband ultrasonic attenuation parameter images (Lefebvre et al., 1998), foetuses and the foetal heads (Pathak et al., 1997), cysts in ultrasound breast images (Yezzi et al., 1997), coronary arteries in intravascular ultrasound images (Sonka et al., 1995) and the pubic arch in transrectal ultrasound images (Pathak et al., 1998).

Many theory-based segmentation techniques exist in the literature, which include wavelet based techniques, morphology based methods, fuzzy clustering based methods, genetic algorithm based methods, artificial neural network based segmentation methods, etc... More details can be found in (Kang et al., 2009; Bovik, 2010; Gao and Xie, 2000). Only two related techniques, fuzzy clustering segmentation and neural network-based segmentation are described bellow.

2.6.4.1 Fuzzy clustering segmentation

The clustering approach that clusters dataset into different clusters of identical attributes has widely been introduced in the field of medical image segmentation in the discrimination of different objects from images (Gonzalez and Woods, 2011). However, the presence of uncertainty in most medical ultrasound imaging data is considered as the main issue that leads to undesirable segmentation outcomes for a specific segmentation approach (Bovik, 2010). This uncertainty and fuzziness between different image classes including boundary regions is mainly caused by the noise, low contrast, and low resolution transducers during image acquisition. In order to overcome the clustering segmentation related problem in image processing, fuzzy-set theory (Zadeh, 1965) can be added to the clustering process to permit fuzzy boundaries to occur between various clusters. In particular, the fuzzy-set theory defines and set-up the concept of uncertainty between classes using a membership fuzzy functions (Zadeh et al., 2014). Fuzzy c-means (FCM) (Bezdek, 1981, 2013) and alternative fuzzy c-mean (AFCM) (Wu and Yang, 2002) clustering segmentation algorithms are two good examples used for MRI segmentation to classify symptomatic and asymptomatic tissues in ophthalmology (Yang et al., 2002) and brain MRI segmentation (Prakash and Kumari, 2017). However, these approaches suffer from some difficulties such as sensitivity to the initialization of the segmentation parameters (number of classes or clusters), the definition of the attribute of fuzzy memberships and computation complexity (Kang et al., 2009; Gao, 2004). In addition to that, most fuzzy clustering segmentation techniques tend to be more suitable for the segmentation of MRI medical images rather than medical ultrasound images when there is clear difference in the variation information of grey-level intensities in the images (Yang et al., 2002).

2.6.4.2 Artificial neural network based segmentation

Artificial neural network (ANN) techniques have attracted considerable attention in medical imaging due to their powerful parallel structure distribution, fast computation, insensitivity to noise effect, intelligence and quick learning capabilities in performing complicated segmentation and classification tasks (Bovik, 2010). ANN is considered as a pixel classification model which can be applied successfully in image segmentation task. Earlier studies (Chang et al., 2010; Noble and Boukerroui, 2006) have proven that integration of ANN can facilitate and improve the segmentation process.

In general, ANNs learn from ground truth samples in the training datasets where their pixels or patterns are already classified manually. The training procedure of the ANNs requires the training of all ANNs parameters and the optimization of all inter-unit connections. As an example, the parameters include, centers of the hidden layer units, the widths of the corresponding activation functions, and the weights between the hidden layers and output layers. The training process stops when all the predicted input-output errors are reduced and the neural network reaches a desired state of accuracy (Agatonovic-Kustrin and Beresford, 2000). In the ANN segmentation process, the feature vectors (extracted and selected from input data) are applied to the ANN through the high dimensionality hidden layer(s) in order to classify the ROIs of medical images. The trained ANN classifies the image pixels into ROI and non-ROI, more specifically, it identifies whether a pixel belongs to ROI or non-ROI. Finally, the results of the image pixels classification are then combined and merged into a region mask (in black and white colour for non-ROI background and ROI foreground, respectively).

There are different important aspects to be considered when designing an ANN based model (Bovik, 2010; Dhawan, 2011): (1) the preparation and the selection of the training dataset samples (ground truth inputs prepared by medical experts using different manual interaction methods), as they should be well represented and distributed; (2) the selection of different useful features (extracted from input datasets) for classification; (3) the topology (structure) of the network and the distribution of the input datasets (training, testing and validation datasets) for classification performance and accuracy; (4) Avoiding over-training during the ANN training process. ANN models such as feed-forward back-propagation (FFBP-NN) and radial basic function (RBF-NN) ANNs models have been widely applied in medical image segmentation (Sarwal and Dhawan, 1998; Ozkan et al., 1993; Dhawan, 2011).

- Feed-forward back-propagation neural network (FFBP-NN)

FFBP-NN (Williams and Hinton, 1986) is the most frequently used paradigm in image processing tasks such as segmentation and classification. A classic feed-forward back-propagation network formed from neurons and organized in layers consists of an input layer, a hidden layer (one or more hidden layers) and an output layer. Every single layer in the above structure is entirely connected to the subsequent layer (where each connection has a weight assigned to it) using a set of interconnected neurons that process the input data in a layered structure. This layered network structure is called a Multi-Layered Perceptron (MLP) which requires a non-linear differentiable activation function (e.g. sigmoidal function) (Zurada, 1992). During the learning stage of the back-propagation network, the real (calculated) output results are compared with the given target results and the error of each input-output pair is calculated. The calculated errors are propagated backwards to the input layer (back-propagation). Finally, the weights of input-output pairs are adjusted and tuned accordingly to predict the correct class labels of the input vectors. This process is repeated until the back-propagation network converges. More details about the structure and the training process of the back-propagation neural network and its applications can be found in (Han et al., 2006; Bovik, 2010; Dhawan, 2011; Zurada, 1992). Despite the popularity and the simplicity of FFBP algorithm in training feed-forwards neural networks, it has some disadvantages (Dhawan, 2011; Priddy and Keller, 2005) such as: (1) greatly affected by the weights initialization and noisy data in the training datasets, which leads to weak generalization performance of FFBP network when classifying new samples, (2) it can be problematic when designing the FFBP-NN topology as it is very hard to find an optimal network topology with an optimal number of hidden layers and nodes in all three layers (input, hidden and output); (3) the choice of the activation function; (4) the learning process can be very slow. As a solution to the above problems, Fahlman and Lebiere (1990) proposed a cascade correlation neural network supervised learning method to define the optimal neural network architecture. Fogel (2000) proposed a different approach to feed-forward training in order to speed-up the training process such as using evolutionary computation to train the neural network weights (Priddy and Keller, 2005).

- Radial basic function neural network (RBF-NN)

RBF-NN (Broomhead and Lowe, 1988; Moody and Darken, 1989) has attracted considerable research interest in the field of pattern recognition and digital image processing due to its functional approximation, interpolation and generalization capabilities (Borş and Pitas, 1999). RBF-NN has been successfully applied in different applications such as: image restoration (Cha and Kassam, 1996), speech recognition (Niranjan and

Fallside, 1990), medical image segmentation (Kovacevic and Loncaric, 1997), and classification (Bishop, 2006), etc. In comparison to FFBP-NN, RBF-NN does not experience sensitivity to the network architecture, and it gives more reproducible, consistent and reliable outcomes (Chen et al., 1991; Jackson et al., 1988; Dhawan, 2011). RFB neural network is designed as a three-layer feed-forward neural network topology: input layer feeding the feature vectors into RBF neural network; hidden layer with radial basis function as activation function and high dimensionality structure; and output layer where all the adjacent layer nodes are fully connected and the linear combination of the hidden weighted radial basis functions are calculated (Orr et al., 1996). The main issues facing RBF network implementation are the location of the centroids and the topology of the RBF-NN. Adaptive k-means and fuzzy clustering methods are two proposed solutions to the aforementioned problems to obtain the optimal number of clusters (Dhawan, 2011).

2.6.4.3 Related work

Although various segmentation methods and techniques for ultrasound images exist, there is not much literature on the segmentation process of the plantar fascia ultrasound images of the foot. The only previous work found in relation to PF tissue US images is that reported in (Deshpande et al., 2013) using the Chan-Vese active contour segmentation method (Chan and Vese, 2001). The Chan-Vese model is based on the variational information in grey-scale intensities of the image. This proposed technique was effective in the detection of bones and in segmenting the soft tissue layers between the bone and the skin in US images of the foot. However, this method is used for segmenting the whole plantar tissue without defining different plantar tissue areas. Most active contour methods used in US images suffer from the following shortcomings that seriously affect the segmentation results (Chang et al., 2010): (1) these methods are sensitive to the environmental noises and edge gradient in the image; (2) they need a clear definition of the initial contour mask; (3) they depend on the number of iterations which may affect segmentation accuracy; and (4) they suffer from a high level of computational complexity. Many researchers have made various improvements to the standard active contour, but the disadvantages of this method are still not fundamentally overcome.

2.6.5 Summary

The plantar fascia US images are significantly different from other US images such as liver, heart, kidney, and abdomen. This is because the structure of the foot PF tissue is visually small in size, and it is located between different layers making the boundaries unclear due to the presence of fatty tissue, nerves and blood vessels. This makes the task of segmenting PF US images in the clinic more challenging; however, this also offers an opportunity for implementing novel approaches to assist the clinicians during diagnosis. Motivated by the advantages offered by the ANN approaches (discussed earlier), in this study we propose an ANN based segmentation method that uses the radial basis function neural network (RBF-NN) classifier to automatically extract and segment the PF area. Different textural features extracted from the region of interest are used and analysed to train the RBF-NN. The trained RBF-NN classifies the PF segments into PF and non-PF region, and then is used to segment the shape of the PF region. Three Different evaluation protocols were used to evaluate the performance of the proposed approach including classification evaluation, segmentation evaluation and statistical evaluation (more details can be found in chapter 4). This method is to our knowledge the first theory based segmentation approach in the plantar fascia US imaging field. Thus the accuracy of this stage is an important step to facilitate the success of the classification and novelty detection process during the clinical diagnosis.

2.7 Feature Extraction techniques

Feature extraction is an important factor and step for different image processing applications such as segmentation based on ANN, pattern recognition, image objects matching and novelty detection. It is mainly required to extract and construct input patterns (i.e. most prominent set of feature vectors that represent various object classes) from ROIs for classifying image object patterns, parameter measurements and image understanding tasks. There are three different types of features to be extracted from medical image objects as summarised in the following subsections and their measures tabulated in Table 2.1 . Their description and mathematical representation can be found in (Loew, 2000; Dhawan, 2011):

2.7.1 Shape features

Shape features are mainly related to the geometric representation of the segmented area such as shape, size and the orientation of the region of interest in the image. These features can be extracted using different measures as summarised in Table 2.1.

2.7.2 Intensity histogram features

Intensity histogram features provide information about the grey-scale distribution of the segmented region and some variations across the region using spatial statistics. It can be calculated using statistical measures as presented in Table 2.1 (Umbaugh, 2005).

2.7.3 Texture features

Texture features are concerned about the spatial arrangement of pixels (i.e. local texture information) within the segmented area or the related objects of the image. Many approaches in the literature exist to extract the texture features. These can be summarised into three different categories: spatial-domain, frequency-domain and model-based methods (Tuceryan and Jain, 1993; Haring et al., 1994; Laine and Fan, 1993; DeKruger and Hunt, 1994).

2.7.3.1 Spacial-based methods

It includes two different types: (1) Grey-level co-occurrence matrix (GLCM) is the the most common statistical method proposed by Haralick (Haralick, 1979). IT takes in consideration the spatial relationship and grey-level first-order distribution of pixels that are determined using certain criterion such as distance, direction or neighbourhood. GLCM features can be extracted using different measures as summarised in Table 2.1 (DeKruger and Hunt, 1994). GLCM can be normalized to produce the second-order GLCM in order to perform better. It can also be extended to carry out volumetric texture analysis of 3-D medical images (Dhawan, 2011). (2) Autocorrelation (or Relational) features can be used to extract information about the offset regularity, recurrent patterns, geometric correlation between objects as well as the fineness of the texture that exist in different direction in the medical image (Chen et al., 2010; Tuceryan and Jain, 1993). However, this method suffers from features redundancy where a large number of irrelevant features are extracted and this will introduce some sort of post-processing step using different feature selection methods to reduce this redundancy (Chen et al., 2010).

2.7.3.2 Spatial-Frequency domain-based

This method uses multi-scale decomposition such as spectral analysis approach which transforms the image into a set of sub-images representing their local texture properties such as frequency and orientation (also called spacio-frequency) elements (Tuceryan and Jain, 1993). There are different spatial-frequency transform approaches for texture retrieval and analysis for extracting energy-based measures presented in table. This includes, DWT (Mallat, 1989), wavelet packet transform (WPT) (Rajpoot et al., 2003), gabor wavelet transform (GWT) (Zhang et al., 2000), DT-CWT (Selesnick et al., 2005) and contourlet transform (CT) (Do and Vetterli, 2005). Although the good effect of these methods on texture retrieval and analysis, some of them are computational expensive and they are still in their early stages of research study and implementation (Dhawan, 2011).

2.7.3.3 Model-based methods

This method is based on the probabilistic models that can represent and synthesise the image textures. It has a set of parameters that controls the definition of the textural properties of the image and features extraction (Chen et al., 2010). As an example, Markov random fields (MRFs) and Gibbs random fields (GRF) are very common methods for modelling images and extracting the local (spatial) contextual features in the desired image. Another example is the Fractals methods which is able to model the statistical quality of coarseness and self-similarity at different image scales (Tuceryan and Jain, 1993). This model suffers from the followings: (1) the difficulty to map a specific texture into the selected probability model; (2) the large numbers of model parameters to be defined; and (2) its popularity (not popular as other texture retrieval methods) (Chen, 2015).

Table 2.1: Feature extraction measures

Feature Extraction Technique	Feature Measures
1) Shape features	Perimeter, area , convexity,projections, circularity, longest and shortest axis, effective diameter, compactness, elongation ratio, hough transform morphological shape descriptors, central moments-based shape etc.
2) Intensity histogram features	mean variance, standard variance, entropy, median intensity, skewness, and kurtosis.
3) Texture features	**Grey-level co-occurrence matrix (GLCM) measures:** contrast, correlation, entropy, energy, homogeneity, mean, variance, median, edge density, inverse difference moment, different variance, information correlation (I and II) sum of square variance, etc. **Spatial-frequency energy-based measures**: norms, mean, variance, standard deviation, etc.

2.7.4 Summary

The main role of feature extraction is to reduce the original data by defining certain characteristics and properties, that distinguish one input pattern from another pattern, and then used as input vectors for classifiers that assign them to the relevant classes that they represent. Concerning the plantar fascia tissue region which has a reasonably defined structure; the most common characteristic of the PF US images is their texture, shape and intensity, and so the main goal of feature extraction in this study is to extract a set of textual features that define the shape of a PF precisely and uniquely and classify different PF ultrasound images into symptomatic and asymptomatic cases. Two different groups of features are defined (according to segmentation and classification approaches). In the proposed segmentation approach six different feature sets were chosen to be extracted from the overlapping patches (32 features for each overlap): Histogram features (Umbaugh, 2005), Haar wavelet features (Wen et al., 2007a), Block-difference of inverse probabilities feature (BDIP) (Chun et al., 2003), Gray level difference statistics (GLDS) (Weszka et al., 1976a), Haralick spatial gray level dependence matrices (SGLDM) (Haralick et al., 1973), Region and shape based features (Gonzalez and Woods, 2011). In this stage, the extracted features are given as input vectors to the RBF neural network to train our data set and classify the PF region (PF and non-PF region) for segmentation process. In the classification approach another six different sets of features are extracted (42 features) including shape features, intensity features , and texture features such as

Neighbourhood Gray Tone Difference Matrix (NGTDM), First Order Statistics (FOS), Statistical Feature Matrix (SFM), Laws Texture Energy Measures (TEM) and GLCM features. In the classification stage, these extracted features are given as input vectors to the neural network classifier to train or test our data set and classify the PF US images to normal or abnormal cases. The features extracted may have some redundancy, thus we need to introduce feature selection and evaluation stage to reduce this redundancy.

2.8 Feature selection techniques

A common problem in most classification processes is that the number of extracted features is much greater than the number of observations (the number of available training samples), which leads to over-fitting deficiency and weak generalization (Yang and Pedersen, 1997). Therefore, feature selection techniques were needed to reduce correlated measurements and to select the most discriminating parameters for improving the generalization efficiency and preventing over-fitting problem. The principal component analysis (PCA) (Jolliffe, 2002) is very common and it is widely used approach for feature selection and dimensionality reduction. However, it is very sensitive to noisy data with sparse distribution (Dhawan, 2011) . The Generic algorithm (GA)-based optimization methods have been proposed to overcome the above PCA problems (Peck and Dhawan, 1993; Dhawan, 2011; Chitre et al., 1994). It has been considered to be more appropriate for non-linear and multi-dimensional data and it is widely used for medical image analysis (Chitre et al., 1994; Huo et al., 2001), but still facing some limitations such as the inconsistency in selectivity pressure and assigning large number of copies to few strings (Dhawan, 2011). A combination of ranking and selection techniques can alter the previous problem. Most of these solutions carry out (sequentially) two functions, ranking and than subset selection (Roffo et al., 2015a). Different examples exist in the literature for feature ranking and selection task including MutInf (Zaffalon and Hutter, 2002), Relief-F (Liu and Motoda, 2007), FSV (Grinblat et al., 2010), SVM-RFE (Guyon et al., 2002), SW Relief-F and SW SVM-RFE (Yu et al., 2012). It has been noted in (Roffo et al., 2015a) that a newly unsupervised proposed method named infinite feature selection (Inf-FS) overcomes all the above approaches in terms of best classification performances measures such as accuracy and average precision. Thus, Inf-FS (Roffo et al., 2015a) approach has been used in this thesis for ranking and selecting the most significant features for segmentation and classification tasks.

2.9 Classification Techniques

Image or selected features classification is the process by which different objects in an image are recognised and characterized according to their texture attributes (Manian et al., 2000; Unger et al., 2015; Dhawan, 2011). As an example, features representing plantar fascia intra-tendinous calcification and perifascial oedema are examined and classified for the detection of foot plantar fasciitis. In order to obtain a perfect classification results, there are three main conditions to be considered during classification process: (1) careful selection of features; (2) a good classifier; and (3) suitable training samples (Unger et al., 2015). There are many possible techniques for image or features classification but most of them fall into the following common categories such as statistical classification methods such as k-nearest neighbour classifier and bayesian classifier, neural network methods such as backpropagation neural network and radial basic function neural network, support vector methods and rule based methods (Dhawan, 2011). In this thesis we minimize our choice to the following common methods: k-nearest neighbor (k-NN), artificial neural network (BPNN and RBFNN) techniques and support vector machines (SVM) methods:

2.9.1 The K-Nearest Neighbor (K-NN) Technique

The k-nearest neighbour algorithm (k-NN) (Fix and Hodges Jr, 1951) is defined as a basic supervised method that uses predefined labelled classes of training examples for classifying objects in various categories depending on the nearest training samples in the feature space (Dhawan, 2011). The k-nearest neighbour algorithm consists of the following steps (Unger et al., 2015; Megalooikonomou et al., 2007):

1. The training phase; where the feature vectors and predefined class labels of the training examples are stored (no learning is performed here, i.e. lazy learning),

2. The classification phase; where the stored features are computed for the test sample (or query, whose class is not labelled) by:

- computing the relative distances between the new vector and the previous stored vectors using different distance metric approaches such as: Minkowsky, Manhattan, Chebychev, Euclidean, Camberra and Kendalls Rank Correlation and selecting *k* (a user predefined constant) nearest samples, and
- predicting (labelling) the new point to fit in the most numerous classes within the training samples set closest to query point.

Several studies have proven that the K-NN is a powerful classification method in many applications including the micro-array classification analysis and human brain classification (Rajini and Bhavani, 2011; Yuan et al., 2004). Although its simplicity and strength, there are still some disadvantages to be considered here such as: (a) the performance of the K-NN is affected by the selection of the user defined constant *k*, which is usually hard to predict it in real domain applications (Yuan et al., 2004); (b) it is computationally expensive and time consuming method when searching for the closest neighbour points; (c) it also requires large storage demands (Guo et al., 2003).

Different proposed solutions exist in the literature (Remus et al., 2008) to overcome the above limitations such as: (1) making the K-NN less reliant on the selection of *k* value by searching at multiple sets of closest-neighbours rather than just one set of closest-neighbours using contextual probability approach (Wang, 2002); (2) reducing the computation time required for finding the closest neighbours using indexing training samples (Cantone et al., 2005). In addition to the aforementioned solutions, it has also been suggested using different feature selection techniques as a pre-processing step to select the most appropriate input features to be used during the process (Yu and Liu, 2004). This will also increase the classification performance and reduces its computation cost. Besides that, choosing the best suitable distance metric approach will also improve the accuracy of K-NN classifiers (Kotsiantis, 2007).

2.9.2 Artificial Neural Networks Techniques

ANNs (also called multilayered perceptrons) (Rumelhart et al., 1988) have been described as intelligent classifiers which use artificial neurons to simulate the biological neurons where their state is characterized and controlled by an active function (Theodoridis and Koutroumbas, 2006). ANNs have powerful capabilities such as intelligent learning from training samples, functional approximation, highly parallel and regular structure, interpolation and good generalization (Duda et al., 2001; Rumelhart et al., 1988; Borş and Pitas, 1999). This is why, they can be seen as a powerful tool for solving various medical image processing and pattern classification problems. For example, they play an important role for detecting liver cancer (Kondo et al., 2011), categorizing the tissue of the placenta (Malathi and Shanthi, 2009), classifying mammogram images (Yun et al., 2006), detecting lung cancer (Taher and Sammouda, 2011), detecting brain tumour (Othman and Basri, 2011), classifying abnormal retinal images (Anitha et al., 2009) and classifying the breast anomalies (Antonie et al., 2001).

There are several ANN techniques used for medical image features and patterns classification. This includes two common used approaches such as backpropagation (BPNN) and radial basic function (RBFNN) which are previously describe in Section 2.6 for ultrasound image segmentation using pixel classification.

ANN is a promising field to apply in the classification of medical images. In spite of this popularity and the large number of applications in the field of medical image analysis, this does not preclude the existence of some shortcomings such as: (a) time consuming during the training phase (going through many passes and nodes to learn the classification rules); (b) the existence of the classification rule articulations due to the structure of the neural network graph and the weights assigned to the links between the nodes; (c) available domain knowledge is rather difficult to be incorporated to a neural network (Lu et al., 1996)

As a solution to the previous ANN problems the theory of Rough Neural Network (RNN) (Yun et al., 2006; Zhai et al., 2007) has been proposed to reduce the original feature sets in the pre-processing classification. The reduction of the original feature sets stage leads to a quicker learning of the neural and better performance (Wang et al., 2007). In addition to that, neuro-fuzzy set approaches (integrating the fuzziness into the decision surfaces using neuro-fuzzy pattern classifier) (Grohman and Dhawan, 2001) have been introduced in the field of ANNs for improving classification and generalization tasks.

2.9.3 Support vector machines (SVM)

Support vector machines (SVM) (Vapnik, 2013) is considered as a supervised learning approach for classifying (linear or no-linear) binary data. SVM is widely used in computational biology for pattern recognition issues (Martínez-Trinidad et al., 2006). SVM splits two classes' samples according to the maximum margin (distance between each set) hyperplane (or decision boundary) search result (Unger et al., 2015). The SVM uses quadratic computing optimization approaches (Cortes and Vapnik, 1995; Vapnik, 2013; Osuna et al., 1997) to solve the related classification problems. Some previous studies (Yang and Liu, 1999) have demonstrated that SVM performed better when compared to other classification methods, such as K-NN, ANN and decision trees, especially when using few training samples and a large number of input variables. This is due to its property mechanism to maximize the margins of class space boundaries which practically improves the generalization ability of the classifier (Yang and Liu, 1999). However, SVM suffers from the following disadvantages (Abe, 2010): (1) it goes through a long training process that demands solving a huge quadratic calculation optimization problem; (2) it is time consuming when selecting an appropriate kernel and its optimal parameters (model selection problem); (3) it uses direct decision functions which make the extension to multi-class problems very hard (needs many formalisations). As a solution to some of SVM shortcomings, different effective approaches have been implemented including: (1) using a sequential minimal optimization (SMO) approach using a fast training algorithm to break the huge quadratic calculation optimization problem into a sequence of smallest problems (handle large training samples) (Platt, 1998); (2) reducing the run-time complexity by using SVM as regression tool and the reformation of the training problem using smaller number of activation basis functions (Osuna and Girosi, 1998); (3) reducing the number of support vectors (requires less parameters and less run-time) by using stochastic acceleration approach or sparse approximation of SVM for solving a kernel lasso technique proposed by Suykens (2017).

2.9.4 Summary

Several works have been carried out using different medical image texture classification methods. But unfortunately, we have not come across of any previous work on classifying plantar fascia ultrasound images. The previous examples describe in general some previous work carried out in terms of general medical images texture classification. There are many possible techniques for texture classification but we minimize our choice to the following common methods: SVM, Linear discriminant analysis (LDA), k-nearest neighbor (k-NN), Decision trees (CART) and radial basic function artificial neural network techniques. In the case of classifying the plantar fascia images depend mainly on the classification model design and algorithms selection. Thus our choice of the classification model over another and the level of classification are decided by investigating and evaluating the five mentioned techniques and choose the best model giving the better results when classifying the plantar fascia images to normal or abnormal. The classification model is evaluated using the confusion matrix and some derived performance measures such as sensitivity, specificity, balanced accuracy, precision, F-score, Matthew's correlation coefficient (MCC), ROC graph and AUC, and time complexity.

2.10 Novelty detection techniques

Novelty detection technique is also known as anomaly detection, outlier detection technique (i.e detecting abnormal samples lying outside the majority of normal samples in the feature space) (Ritter and Gallegos, 1997) or one-class classification technique (only normal data used during training phase) (Moya and Hush, 1996). Novelty detection can be introduced as a binary classification task that distinguishes in certain respects between test data samples (abnormal data not available during training) and the initial normal data samples (available during training). It is mainly applied to datasets in which a large number of normal samples exist and where there is a lacking in datasets to describe abnormality behaviour. Novelty and anomaly detection approaches have been widely involved in many application fields such as medical diagnostic issues (Quinn and Williams, 2007), detecting masses in mammograms (Tarassenko et al., 1995), structural damage (Surace and Worden, 2010), text data mining (Basu et al., 2004), fault detection (King et al., 2002), and others. In this thesis we are only concerned with medical image processing novelty detection domain. The medical imaging data can have anomalies due to several causes such as abnormal patient condition and characteristics or machine faults or data acquisition errors. Some of novelty detection approaches dealing with this kind of data can be categorized in three main groups such as probabilistic-based, Neighborhood-based, classification-based novelty detection techniques (Campilho and

Karray, 2016).

2.10.1 Probabilistic-based novelty detection

Probabilistic-based novelty detection techniques are mainly based on the density estimation of the normal data using a generative probability density function (PDF) (i.e low density regions in the training dataset shows that these regions have a small chance of containing normal samples) (Pimentel et al., 2014; Campilho and Karray, 2016). There are different generative methods used in the literature to calculate the PDF for implementing normal data models including Gaussian mixture model (Shental et al., 2003), theoretical support (Park et al., 2010) and no-linear projection pursuit (Breaban and Luchian, 2012). The strength of probabilistic-based novelty detection is represented by its strong mathematical formation and the minimum requirement of the amount of information. However, its performance is affected when using small amount of training samples (Pimentel et al., 2014).

2.10.2 Neighborhood or distance-based approach

Neighbourhood-based approach is considered as the most commonly used method for novelty detection. It is mainly based on the nearest-neighbour and clustering analysis conceptions (as in classification tasks). The idea behind these conceptions is that normal dataset samples are tightly grouped together, while new dataset (anomalies or novel data) appears outside or too far from their nearest neighbours (Pimentel et al., 2014). This approach uses three different methods in order to define the novelty score (also called threshold) (Chandola et al., 2009): (1) methods using distance measures (such Euclidean (Tan et al., 2005), Minkowski, Manhattan and Mahalanobis measures for continuous data, and distance simple matching coefficient measures for categorical data (Boriah et al., 2008; Chandola et al., 2008)) between data entity and its K^{th} closest neighbour entity; (2) methods that calculate the relative density of each data entity to find its novelty threshold (i.e. data points belonging to low density neighbourhood are treated as anomalous wile other data points that belong to dense neighbourhood are treated as normal) (Chandola et al., 2009); and (3) methods using clustering algorithms such as k-means, c-means, fuzzy c-means and possibilistic c-means (Jain and Dubes, 1988; Tan et al., 2005) for grouping similar (normal) data points into different clusters while novel data does not belong to any cluster (Chandola et al., 2009). Neighbourhood-based techniques share some common features with probabilistic techniques but do not require any previous knowledge of the data distribution (i.e. it is unsupervised in nature) (Campilho and Karray, 2016). However this approach suffers from the following effects: (1) it is computationally expensive when calculating the closest neighbour between data entities (for testing or training data); (2) its performance depends significantly on the distance and similarity measures calculation; (3) it is hard to define these measures between data entities; (4) in case of clustering approach, it is difficult to select the right value of cluster width for complex data samples; and (5) it is not flexible enough to define local novel data entities in datasets that have varied densities and arbitrary forms (Chandola et al., 2009; Campilho and Karray, 2016; Pimentel et al., 2014).

2.10.3 Classification based novelty detection techniques

Classification based novelty detection techniques use two main stages, training stage to train and learn the classifier (model) from an existing training samples (labelled datasets) and testing stage to classify the testing samples as normal or novel (anomalous) data using the previously trained classifier. The following subsection describe only two classification based novelty detection techniques such as neural networks and support vector machines. These techniques are widely applied to novelty detection in the one-class setting where all the training samples have only one class label (e.g. normal) (Chandola et al., 2009; Campilho and Karray, 2016; Pimentel et al., 2014).

2.10.3.1 Neural network novelty detection technique

In general, one-class or multi-class novelty detection techniques using neural networks are performed in two different stages: stage one, neural network is trained using normal training data (to learn normal instances), stage two, the input testing instances are fed to the trained neural network model. These inputs are either accepted or rejected for normal or novel data detection, respectively (De Stefano et al., 2000). Various types of neural networks have been used in the literature for novelty (anomaly) detection (Chandola et al., 2009) including, multi layered perceptrons (back-propagation neural network) (Augusteijn and Folkert, 2002), neural trees (Martinez, 1998), auto-associative networks (Aeyels, 1991), adaptive Resonance theory based (Moya et al., 1993), radial basis function (Bishop, 1994; Albrecht et al., 2000), hopfield networks (Jagota, 1991) and oscillatory networks (Ho and Rouat, 1997; Tuong Vinh and ROUAT, 2001). As an advantage, the testing stage of the classification techniques is processed quickly since the testing samples (attributes) are compared with the trained classification model. However, the main drawback of this group is assigning a non desired label to each testing attribute (i.e. assigning meaningful novel (anomaly) score for testing attributes) (Chandola et al., 2009). As a solution to this, a probabilistic prediction score approach (in modifying SVMs to produce probabilities) was proposed to predict the novel score from the outcomes of the desire classifier (Platt et al., 1999).

2.10.3.2 One-class SVM-based novelty detection technique

The Support Vector Machines (SVMs) are well-known classification technique used to search for an optimal hyperplane to split up samples (attributes) into various classes (Vapnik, 2013). SVMs have been widely applied to different novelty and anomaly detection applications using one-class approach (Ratsch et al., 2002; Martinez, 1998; Ma and Perkins, 2003). The One-class SVM method proposed by Schölkopf (1999) presumes that class instances lying outside of the class boundary (area that contains the training data instances) are treated as novel (or anomalous) and the model is built based on the boundary of training (normal) data. For this reason, different kernel functions can be used to learn complex class areas including radial basis function (RBF) kernel. The drawbacks of these techniques can be summarised as follows (Pimentel et al., 2014): (1) most of the kernel functions used are computationally complex; (2) it is difficult to choose an appropriate kernel function; and (3) it is also hard to set up its parameters and select the appropriate threshold (the size of the boundary areas, usually selected empirically). Different approaches have been proposed to overcome the previous shortcomings such as:, (1) using the support vector data description (SVDD) method, (proposed by Tax and Duin (1999b)), that introduces automatic model parameters optimisation method; (2) using different kernels with linear computing optimisation methods (instead of the quadratic one) typically applied with SVMs (Campbell and Bennett, 2001); other proposed approaches (mainly based on either SVDD or one-class SVM methods) can be found in (Pimentel et al., 2014).

2.10.4 Summary

Detecting novelty and anomalies in medical image processing data domain is a challenging problem for researchers. There are various applications dealing with medical images for novelty (anomaly) detection in the literature. However, most of these applications tend to be model dependent due to the following aspects: (1) the definition of novelty or anomaly concepts (this definition is mainly related to a specific problem), (2) data description (e.g. nature and size of the datasets), (3) the challenges aspect facing novelty detection process, and (4) the anomaly detection approaches used. Therefore, a suitable model for novelty or anomaly detection in PF US images is required. In this study we propose a one-class model using SVDD method, which is applied to the normal plantar fascia ultrasound images dataset for defining plantar fascia abnormalities behaviour (i.e. identifying novelty scores or thresholds). The effectiveness of the one-class novelty model is evaluated and compared to GMM, PARZEN, GPOC and SOM models using balanced accuracy, F-score, MCC, Gmean, AUC and Time cost.

Chapter 3

Speckle Noise Reduction In Ulrasound Imaging of the Plantar Fascia: A Comparative Evaluation

3.1 Overview

This chapter introduces a comparative evaluation study of seven groups and a total of 16 speckle-reduction methods that were applied to the US images. This comparative study is mainly based on different evaluation protocols such as: image quality evaluation metrics, feature extraction and selection analysis, and visual evaluation by clinical experts in the assessment of 286 PF US images. The following Section 3.2 gives a brief introduction about speckle noise effect, filtering methods and the scope of the proposed evaluation study. Section 3.3 describes materials and PF US image acquisition. Section 3.4 summarises different despeckling methods used in this study along with their parameters settings. Section 3.5 introduces the evaluation and performance protocol used to assess the filtering methods. Section 3.6 presents and discusses the experimental findings. Finally, Section 3.7 summarizes our findings and concludes this study with a brief discussion on the scope for future work.

3.2 Introduction

Although US imaging offers many advantages, it produces low-quality images owing to the presence of speckle noise during the process of images acquisition (Jain and Tyagi, 2014; Ganzalez and Woods, 2002). The effect of speckle noise is very common in US images, it reduces the image contrast (Saraniya and Ezhilarasi, 2014; Burckhardt, 1978), thereby destroying or diffusing the image edges, and making medical interpretation and biometric measurements more difficult during diagnosis. In addition, speckle noise limits the efficiency of some US image processing applications such as edge detection; automated segmentation; feature extraction, reduction, ranking and selection; and image registration. Therefore, many speckle reduction post-processing methods have been proposed in the literature (Finn et al., 2011; Zhang et al., 2015; Loizou et al., 2014a; Loizou and Pattichis, 2008) (as reported earlier in Section 2.4) to suppress or reduce speckle noise in US images. A thorough review of all these speckle reduction methods is beyond the scope of this project. For a wide range of denoising and despeckling techniques, a recent quantitative comparative study of 48 filters can be found in (Biradar et al., 2015). In this study, we carried out a proposed comparative evaluation study of some common existing speckle-reduction methods (7 groups and 16 filters as summarized in Section 3.4). This evaluation is based on qualitative evaluation metrics, feature extraction and selection analysis, and visual evaluation by two medical expert in the assessment of 286 normal and abnormal plantar fascia US images.

3.3 Materials and PF US image acquisition

Different plantar fascia US images, scanned from a patient's footprint area (in the prone position with free ankle motion) were used in this study. More specifically, 286 different real US images (250 normal and 36 abnormal) were collected from 45 patients to compare the presented methods (for different PF structures rearfoot, midfoot and forefoot sections) with 256 gray levels, a size dimension of 600×655 pixels and a resolution of 28.35 pixels/cm. All these images were obtained from the Health Sciences Department, University of Salford, directly collected from the US device and acquired by two professional clinicians through a portable Venue 40 musculoskeletal US system (GE Healthcare, UK) with a $5-13$ MHz wideband linear array probe $(12.7\ mm \times 47.1\ mm)$.

Depending on the position and the orientation of the probe, the US images acquired might not have the expected results and can be defective, making medical interpretation and biometric measurements more difficult during diagnosis. This is why all the PF scans were performed separately by the operator according to the general medical scan protocol as in (Crofts et al., 2014) to obtain the best possible PF US images (Figure 3.1).

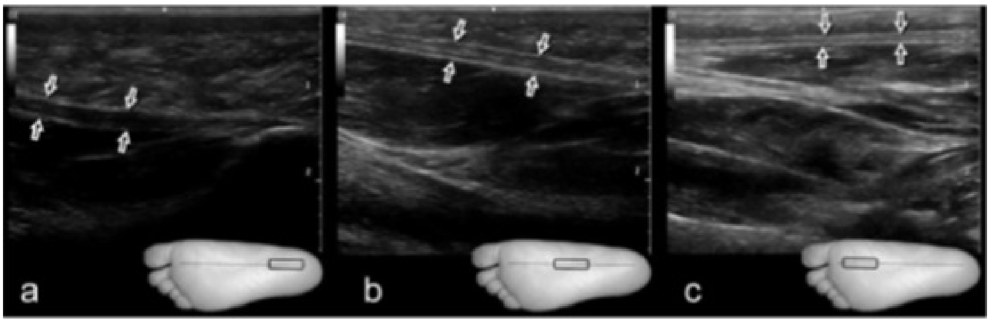


Figure 3.1: Probe position, longitudinal orientation and sample US images for all PF different structures. (a) Rearfoot PF section; (b) Midfoot PF section; and (c) Forefoot PF section

.

3.4 Despeckle filtering methods

As discussed previously in Section 2.2 the speckle noise in PF US images is a multiplicative in nature and can be generalized as in equation (2.2) . In the operation of transforming the speckle noise (multiplicative) into the classical additive noise, the logarithmic transform was performed using equation (2.3) . Seven groups and a total of 16 despeckling methods were investigated and applied on the 286 logarithmically transformed PF US images. Their mathematical background, advantages and disadvantage were previously discussed in more details in Section 2.4 . However, most of these filters are very sensitive to changes of their parameter values and settings. Therefore, to get good filtering results, these parameters values are chosen based on available discussions in research studies (Zhang et al., 2015; Loizou et al., 2014a), some despeckling experiments' results presented in (Biradar et al., 2015) and medical experts. The following tables 3.1 and 3.2 summarise all the despeckling filters used in this study and their parameters values, respectively.

Table 3.1: An Overview of different speckle reduction techniques used in this study

Despeckling technique	**Filter name and references**	**Filter concept**
1) Median-based filtering	Median (Loupas et al., 1989)	Based on median filtering (change the center pixel value in the kernel by its median-value of its neghbour) .
2) Local statistics-based	Mean variance (MeanV) and Wiener (Loizou et al., 2002; Jain, 1989)	Based on sliding kernel window and local statistics information.
3) Homogeneity-based	Homog (Christodoulou et al., 2002)	Based on the computation of the most homogeneous neghborhood area arround each pixel.
4) Geometric-based	Geom (Busse et al., 1995)	Based on no-linear geometric iterative calculation.
5) Anisotropic diffusion-based	PMAD (Perona and Malik, 1990), SRAD (Yu and Acton, 2002) and DPAD (Aja-Fernández and Alberola-López, 2006a)	Non-linear despeckling methods, performing both speckle denoising and contrast enhancement using diffrent diffusion approaches.
6) Wavelet-based filtering	DWT, DT-CWT, DT-CWT_S , DT-CWT_H, DT-CWT_T, DT-CWT_B (Rabbani et al., 2008) (Michailovich and Tannenbaum, 2006)	Standard and complex wavelet-based filters using different thresholding methods to eliminate unuseful or noisy wavelet coefficients.
7) Hybrid filter	DLWFDW (Shui, 2005b)	Integrating adaptive local statistics Wiener filtering in the wavelet transform domain.
	HybridMedian (Nieminen et al., 1987) (Loizou and Pattichis, 2008; Loizou et al., 2014a)	An improved version of Median filtering. It computes the median of the filtering results generated by the median filter using two kernel shapes (x and +) to preserve edges and improving despeckling process

Table 3.2: Parameter setting values for each despeckling method

Filter name	Parameter setting values
Median	kernel window size = 5 x 5 with 3 iterations
MeanV	kernel window size = 5 x 5 with 2 iterations
Wiener	kernel window size = 5 x 5 with one iteration
homog	kernel window size = 5 x 5 with 3 iterations
geom	kernel window size = 5 x 5 with 5 iterations
PMAD	Diffusion constant = 30, Diffusion rate = 0.25, iterations = 5, 20
SRAD	Iterations = 5, 30 time step = 0.02, $\rho = 1$
DPAD	Iterations = 5, 30 time step = 0.02 with Cu noise calculation
DWT	Kernel window size 5 × 5, decomposition stages J = 2, threshold T = 20, soft thresholding
DT-CWT	Kernel window size 5 × 5, decomposition stages J = 2, threshold T = 20, soft thresholding
DT-CWT_S	Kernel window size 7 × 7, decomposition stages J = 4, BayesShrink rules, soft thresholding
DT-CWT_H	Kernel window size 7 × 7, decomposition stages J = 4, BayesShrink rules, hard thresholding
DT-CWT_T	Kernel window size 7 × 7, decomposition stages J = 4, BayesShrink rules, trimed thresholding
DT-CWT_B	Kernel window size 7 × 7, decomposition stages J = 4, BayesShrink rules, bivariate thresholding
DLWFDW	Kernel window size 7 × 7, decomposition stages J = 5, wavelet based db4, wiener filtering
HybridMedian	kernel window size = 5 x 5 with 2 iterations

According to the analysis of the wavelet based despeckle filtering methods discussed previously, there were three implementations: (1) DWT and DT-CWT implementation using universal VisuShrink rules and soft thresholding function; (2) DT-CWT_S, DT-CWT_H, DT-CWT_T and DT-CWT_B implementation using BayesShrink rules and different sub-band thresholding functions and DLWFDW implementation. The following steps summarise their implementations:

1. Apply log transformation to the US images.

2. Decompose the log-transformed filtered US image, (using the forward wavelet based filters on window size of (5x5), (7x7) and over scales $J = 2$ and $J = 4$, into details and approximate image sub-bands based on wavelet based methods (DWT, DT-CWT, DT-CWT_S, DT-CWT_H, DT-CWT_T, DT-CWT_B).

3. When using DLWFDW apply DWT or UDWT wavelet decomposition and the first filtering stage using wiener filter as in (2.40) with three orientation windows (horizontal, vertical and diagonal), followed by wavelet composition to get the partially filtered image.

4. Implement the universal VisuShrink rules for standard DWT, and complex DT-CWT as follows:

   (a) Calculate the standard deviation of the noise σ in each sub-band according to Donoho and Jhonstone's method, given by (2.33);

   (b) Estimate the various thresholds T in the various wavelet sub-bands using (2.32);

5. Apply the subband adaptive BayesShrink rules for DT-DT-CWT_S, DT-CWT_H, DT-CWT_T, DT-CWT_B and DLWFDW as follows:

   (a) Compute the noise variance σ_D according to Donoho and Jhonstone's method, given by (2.36);

   (b) Calculate the variance σ_n of the wavelet coefficients in the various sub-bands W_s using (2.39);

   (c) Calculate the signal variance σ of each wavelet sub-band using (2.37);

   (d) Estimate the various thresholds T_s in various wavelet sub-bands using (2.35);

6. Apply only soft thresholding and shrinkage method to DWT and DT-CWT (using VisuShrink rules) coefficients for all sub-bands and scales;

7. Apply different thresholding and shrinkage methods to the DT-CWT (using BayesShrink rule) coefficients for all sub-bands and scales using the following shrinkage methods: (a) Hard thresholding using (2.26), (b) Soft thresholding using (2.27), (c) Trimmed thresholding using (2.28), (d) Bivariate thresholding using (2.31).

8. Apply the second filtering stage of DLWFDW using the second wiener filter.

9. Apply the inverse wavelet transform (DWT, DT-CWT and DLWFDW) to the estimated coefficients.

10. Perform exponential transformation to obtain the despeckled image.

3.5 Performance and evaluation protocol

Speckle reduction filters are applied directly to two different sets of images: (1) original unfiltered clinical raw PF US images that are extracted directly from the US machine without any modification, and (2) modified US images using a simulated speckle noise with a variance of 0.05. To judge the performance of the selected filters in terms of edge preservation, the ability to retain tiny important medical details, and effective speckle reduction, three evaluation protocols were used in this study such as, image quality evaluation metrics, feature extraction and selection analysis and visual evaluation by medical experts. In addition to this, it aims to highlight the effect of multiplicative speckle noise on the global image quality improvement and edge preservation, by introducing a simulated speckle on the original images.

3.5.1 Image quality evaluation metrics

Nine different well known quantitative evaluation metrics were used for calculating the difference between each pixel pair in the original and despeckled US images. These filters are: mean square error (MSE), root mean square error (RMSE), signal-to-noise ratio (SNR), peak signal-to-noise ratio (PSNR), error summation 3 (ERR3), error summation

4 (ERR4), universal quality index (UQI), structural similarity index map (SSIM), average difference (AD), correlation coefficient (CoC), edge preservation index (EPI). Their mathematical representations and significances are summarised bellow and discussed in more details in (Wu et al., 2013; Sivakumar et al., 2010; Wang et al., 2004; Gupta et al., 2007; Srivastava et al., 2010).

3.5.1.1 Mean square error

MSE measures the quality change (average difference) between the original image and the despeckled image; it is given by:

$$MSE = \frac{1}{M \times N} \sum_{j=1}^{M} \sum_{k=1}^{N} \left(X_{j,k} - X'_{j,k} \right)^2, \tag{3.1}$$

where $M \times N$ is the image size, while $X_{j,k}$ and $X'_{j,k}$ denote the gray values of the original and despeckled images at points j,k, respectively.

3.5.1.2 Root mean square error

RMSE represent the result of the square root of the squared average error upon a pixel kernel window (Loizou et al., 2006). It is calculated as:

$$RMSE = \sqrt{\frac{1}{M \times N} \sum_{j=1}^{M} \sum_{k=1}^{N} \left(X_{j,k} - X'_{j,k} \right)^2}. \tag{3.2}$$

Lower MSE and RMSE values indicate the filtered image is closer to the ground truth (reference) image and zero indicating equality.

3.5.1.3 Signal-to-noise ratio

SNR is the ratio between the ground truth image and the despeckeled image. It is given by:

$$SNR = 10 \log_{10} \left(\frac{\sum_{j=1}^{M} \sum_{k=1}^{N} \left(\left(X_{j,k} \right)^2 - \left(X'_{j,k} \right)^2 \right)}{\sum_{j=1}^{M} \sum_{k=1}^{N} \left(X_{j,k} - X'_{j,k} \right)^2} \right). \tag{3.3}$$

3.5.1.4 Peak signal-to-noise ratio

PSNR measures the maximum power of the the ground truth and despeckled image (Sivakumar et al., 2010). It is statistically represented by:

$$PSNR = 10\log_{10}\left(\frac{(2^n-1)^2}{MSE}\right), \tag{3.4}$$

where n is the number of bits used to represent an image pixel and MSE is the mean square error defined previously. For grey scale images with $n = 8$, equation (3.4) can be rewritten as:

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right). \tag{3.5}$$

Large SNR and PSNR values demonstrate efficient speckle reduction filter and higher image quality. The PSNR becomes undefined for similar images (Sivakumar et al., 2010).

3.5.1.5 Error summations: ERR3 and ERR4

The error summations ERR3 and ERR4 (the norm of dissimilarity between the reference and the filtered images) (Wang et al., 2004; Loizou et al., 2005) are computed using the following Minkowski metric for $\beta = 3$ and $\beta = 4$, respectively.

$$ERR = \left(\frac{1}{M\times N}\sum_{j=1}^{M}\sum_{k=1}^{N}\left|X_{j,k}-X'_{j,k}\right|^{\beta}\right)^{1/\beta}. \tag{3.6}$$

3.5.1.6 Universal quality index

UQI measures the degree of distortion (based on lack of contrast, correlation and luminance) between the ground truth and filtered image. Its values vary between -1 and 1. For similar images, the best value of the image quality index equal 1. UQI is defined as:

$$UQI = \frac{4\sigma_{XX'}\left(\bar{X}\bar{X'}\right)\left(\sigma_X\sigma_{X'}\right)}{\sigma_X\sigma_{X'}\left(\bar{X}^2+\bar{X'^2}\right)\left(\sigma_X^2+\sigma_{X'}^2\right)}, \tag{3.7}$$

where $\bar{X}$ and $\bar{X}'$ are the mean average values of the original image X and the denoised image X', respectively, $\sigma_{XX'}$ is the standard deviation (covariance between the original image and the despeckled image), and σ_{X^2} and σ'^2_X are the variances of X and X', respectively. In (Wang et al., 2004) has been noted that the UQI is not useful for smooth US images. Therefore, it has been replaced by a generalized version of the UQI, namely, the structural similarity index map (SSIM).

3.5.1.7 Structural similarity index map

SSIM measures the structural similarity between the original image and the despeckled image. It is given by:

$$SSIM = \frac{\left(2\bar{X}_{i,j}\bar{X}'_{i,j} + C_1\right)\left(2\sigma_{XX'} + C_2\right)}{\left(\bar{X}^2_{i,j} + \bar{X}'^2_{i,j} + C_1\right)\left(\sigma^2_X + \sigma^2_{X'} + C_2\right)}, \tag{3.8}$$

where C_1 and C_2 are constants (equal to 2.55 and 7.65 dr, respectively, where dr is the dynamic range of the intensity). The SSIM values vary between -1 and $+1$. For identical images and better visual quality, the SSIM should be close to one.

3.5.1.8 Average difference

AD represents the ratio value between the mean difference (between the the ground truth and filtered image) and the image size. Small AD values indicate similar images and effective despeckling filter (Rosa and Monteiro, 2014). AD is mathematically calculated as:

$$AD = \frac{1}{M \times N} \sum_{j=1}^{M} \sum_{k=1}^{N} \left| X_{j,k} - X'_{j,k} \right|. \tag{3.9}$$

3.5.1.9 Correlation coefficient

CoC also measures similarity degree between the reference image and speckel reduced image. Its values range between 1 and 0 for similar and dissimilar images, respectively. (Sivakumar et al., 2010). It is defined as

$$CoC = \frac{\sum_{j=1}^{M}\sum_{k=1}^{N}\left(X_{j,k}-\bar{X}_{j,k}\right)\left(X'_{j,k}-\bar{X}'_{j,k}\right)}{\sqrt{\sum_{j=1}^{M}\sum_{k=1}^{N}\left(X_{j,k}-\bar{X}_{j,k}\right)^{2}\sum_{j=1}^{M}\sum_{k=1}^{N}\left(X'_{j,k}-\bar{X}'_{j,k}\right)^{2}}}, \tag{3.10}$$

where $\bar{X}$ and $\bar{X}'$ are the means of the original image and the despeckled image, respectively.

3.5.1.10 Edge preservation index

EPI measures the level of edge preservation in the despeckled image. For best image quality EPI equal to one (Rabbani et al., 2008).

$$EPI = \frac{\sum_{j=1}^{M}\sum_{k=1}^{N}\left(\Delta X_{j,k}-\overline{\Delta X}_{j,k}\right)\left(\Delta X'_{j,k}-\overline{\Delta X'}_{j,k}\right)}{\sqrt{\sum_{j=1}^{M}\sum_{k=1}^{N}\left(\Delta X_{j,k}-\overline{\Delta X}_{j,k}\right)^{2}\sum_{j=1}^{M}\sum_{k=1}^{N}\left(\Delta X'_{j,k}-\overline{\Delta X'}_{j,k}\right)^{2}}}, \tag{3.11}$$

where ΔX and $\Delta X'$ are the high-pass filtered X and X', respectively, using the discrete Laplacian operator (a 3×3 pixel standard approximation), while $\overline{\Delta X}$ and $\overline{\Delta X'}$ are the means of ΔX and $\Delta X'$, respectively. For US medical applications, the computed values of UQI, SSIM, CoC, and EPI should be close to unity for effective despeckling, and thus, high diagnostic quality.

3.5.2 Feature extraction analysis

Feature extraction is an important step in defining useful information about the PF characterization in US images. It aims to extract the most prominent features that represent various object classes in PF US images. In this study, 6 different feature sets (a total of 33 features) were extracted both from speckled and filtered PF US images. Their mathematical representation can be found in the referred citations.

3.5.2.1 Histogram features

Histogram features also known as first order features are defined as a statistically based features, where the histogram is used as a model of the probability distribution of the gray levels (Umbaugh, 2005). They measures the gray level distribution texture characteristics of an image or a sub-image. The first order histogram probability feature is defined as follows:

$$P(g) = \frac{N(g)}{M}, \tag{3.12}$$

where M is the number of pixels in the image, $N(g)$ is the number of pixels at gray level g. In this study, we considered most the histogram features including (1) mean, (2) standard deviation, (3) skewness, (4) energy, and (5) entropy (Umbaugh, 2005).

1. **Mean:** calculates the general brightness of the image, high and low values represent brighter and darker image, respectively. The histogram mean feature can be defined using the following measure:

   $$\bar{g} = \sum_{g=0}^{L-1} gP(g) = \sum_{r}\sum_{c} \frac{I(rc)}{M}, \tag{3.13}$$

   where L is the total number of the gray levels which range from 0 to $L-1$, r and c are rows and columns corresponding to the pixels in the image I.

2. **Standard Deviation (sd):** defines the contrast of the US image, so high and low values represent high and low contrast image, respectively. It has been calculated using:

   $$\sigma_g = \sqrt{\sum_{g=0}^{L-1} (g-\bar{g})^2 P(g)}. \tag{3.14}$$

   where g and $\bar{g}$ represent the gray level values and the mean value.

3. **skewness (skew):** describes the asymmetry about the mean in the gray level distribution. The skewness can be measured using:

   $$Skew = \frac{1}{\sigma_g^3} \sum_{g=0}^{L-1} (g-\bar{g})^3 P(g). \tag{3.15}$$

4. **Energy (eng):** defines how the gray levels are distributed. It can be calculated as follows:

$$Energy = \sum_{g=0}^{L-1} [P(g)]^2. \tag{3.16}$$

5. **Entropy (ent):** calculates the number of bits required for coding the image data. It can be defined using:

$$Entropy = -\sum_{g=0}^{L-1} P(g) \log_2 [P(g)]. \tag{3.17}$$

.

3.5.2.2 Haar wavelet features

Haar wavelet features extract edges and object shape information from image multi-scale transformation (Wen et al., 2007b). They have been widely applied in many areas such as image retrieval, objects detection and face recognition (Wen et al., 2007a). In this work mean (**`h_mean`**) and variance (**`h_var`**) haar wavelet features (Gonzalez and Woods, 2011) of the low-low (LL) frequency subband (LL band denotes approximation details) were calculated using equations (3.18) and (3.19), respectively.

$$\texttt{h\_mean} = \mu_{x,y} = \frac{1}{M^2} \sum_{(x,y)\in B} I(x,y), \tag{3.18}$$

$$\texttt{h\_var} = \sigma^2_{x,y} = \frac{1}{M^2} \sum_{(x,y)\in B} (I(x,y) - \mu_{x,y})^2, \tag{3.19}$$

where $I(x,y)$ is the intensity of a pixel (x,y) in the region of interest block after the haar wavelet decomposition, and B is the block size of $M \times M$.

3.5.2.3 Gray level difference statistics

Gray level difference statistics (GLDS) (Weszka et al., 1976a) method extracts the following five features: (1) contrast (cont), (2) homogeneity (hom) , (3) entropy (ent), (4)energy (eng) and (5) mean, using first-order local statistics values based on absolute differences between pairs of gray levels. In this work, GLDS were computed for the following displacements: $\delta = (0,1),(1,1),(1,0),(1,-1)$, where $\delta \equiv (\Delta x, \Delta y)$ and their average values were calculated.

3.5.2.4 Haralick spatial gray level dependence matrices

Spatial Gray Level Dependence Matrices (SGLDM) represent the most popular statistical feature extraction method proposed by Haralick (1973), and it considers spatial relationship of image pixels. It is calculated using the second-order joint conditional probability density functions (PDFs) with respect to two parameters: inter-pixel distance d and direction angle θ. In this study, 13 SGLDM features were calculated and averaged for a selected distance $d = 1$ (3×3 matrices) and four different orientation angles $\theta = 0^o, 45^o, 90^o$, and 135^o. The computed SGLDM features were: (1) angular second moment (asm), (2) contrast (cont), (3) correlation (corr), (4) sum of squares (sum_sq), (5) variance(var), (6) inverse difference moment (inv_dif_mom), (7) sum average (sum_avg), (8) sum variance (sum_var), (9) sum entropy (sum_ent), (10) entropy (ent), (11) difference variance (diff_var), (12) difference entropy (diff_ent), and (13) information measures of correlation (inf_m_corr).

3.5.2.5 Fourier power spectrum

In the frequency domain discrete 2-D Fourier transform (FT) of an image can be defined by

$$F(u,v) = \frac{1}{N^2} \sum_{x,y=0}^{n-1} e^{-2\pi\sqrt{-1(xu+yv)}} f(x,y), \ 0 \leq u,v \leq n-1, \tag{3.20}$$

and the Fourier power spectrum $(FPS) = |F|^2 = FF^*$, where * represents the complex conjugate (Lendaris and Stanley, 1970; Weszka et al., 1976a; Wu et al., 1992). In this study the radial sum distribution and the angular sum distribution of the FPS are computed using the ring-shaped samples equation (3.21) and the estimation of wedge-shaped samples equation (3.22), respectively,

$$\phi_{r_1 r_2} = \sum_{\substack{r_1^2 \leq u^2+v^2 < r_2^2 \\ 0 \leq u,v \leq n-1}} |F(u,v)|^2, \tag{3.21}$$

$$\phi_{\theta_1 \theta_2} = \sum_{\substack{\theta_1 \leq tan^{-1} + (v/u) < \theta_2 \\ 0 < u,v \leq n-1}} |F(u,v)|^2, \tag{3.22}$$

where $\phi_{r_1 r_2}$ and $\phi_{\theta_1 \theta_2}$ are the radial and the angular distributions of the FPS; $F(u,v)$ and $|F(u,v)|^2$ represent the FT and FPS of the $N \times N$ image, respectively; r_1, r_2 denote the inner and outer radii of the ring, respectively; and θ_1, θ_2 are the lower and upper limit of the wedge, respectively.

3.5.2.6 Region based features

Region based features are mainly related to shape, size and orientation of the ROI in the image. The following seven regional features were computed from the PF ROI: (1) area, (2) perimeter (perim), (3) major axis length (maj_ax_len), (4) minor axis length (min_ax_len), (5) equivalent diameter (equ_diam), (6) extent (ext), and (7) convex area (conv_area).

3.5.3 Feature selection and ranking analysis

Feature ranking and selection techniques were introduced to reduce correlated measurements and to select the most discriminating parameters for improving the efficiency of the despecling methods. In this study an unsupervised filter-based feature selection method called infinity feature selection method (Inf-FS) proposed by Roffo et al. (2015b) was introduced for weighting and ranking different features (33 features) extracted both from unmodified original images (or simulated speckeled images) and filtered images (a total of 286 PF US images). This technique has frequently been used in previous work for feature ranking and selection in classification problems with ultrasound images (Roffo et al., 2015b). It uses the convergence properties of power series of matrices, and applies the infinite feature selection concept in order to examine the significance (weights) of different feature sets and rank their importance order accordingly; for more details and its mathematical representation can be found in (Roffo et al., 2015b). Feature ranking and weights selection scoring test was carried out here to check if a significant difference exist between features (weights and ranking orders) calculated on the original and the filtered US PF images. It should be noted here that, for all extracted features, a higher feature weight and an improved ranking order after using different despeckling methods show filtering improvements. The best filtering methods are the ones with the highest ranking order scores.

3.5.4 Visual evaluation by medical experts

Following ethical approval from the School of Computing, Science and engineering, University of Salford (ST1617-48), voluntary medical observers were requested to perform the relative visual grading analysis (VGA) (Almen et al., 2000) in order to evaluate the quality of 120 PF US images using 16 different despeckling methods. The grading analysis was carried out using Bespoke software (Hogg and Blindell, 2012) with dual side-by-side 5 megapixel calibrated monitors (The Royal College of Radiologists, 2014) and dimmed ambient lighting (< 10 lux), being agreed with normal image conditions (Perry et al., 2006). The monitors calibration was performed according to the digital imaging and communication in medicine greyscale standard display values (National Electrical Manufacturers Association (NEMA), 2011). The observers consisted of two medical experts with more than 10 years experience in ultrasound imaging and feature tracking for skin, muscles of the lower limb, foot, ankle and plantar tissue. The experts were blinded to the despeckling filters and the parameter values used, and they were required to assign a score in the one-to-five point Likert scale (1=much worse, 2=fairly worse, 3=about the same, 4=better, 5=much better than the original unfiltered image). The scores were assigned to three different criteria items including: (1) global image quality improvement; (2) plantar fascia region definition; and (3) the sharpness of the plantar fascia edges. The experts are allowed to do equal scoring for more than one image in each class and filter, the mean score will be calculated. The experts need to evaluate the area around the PF (Plantar Fascia) and examine the inner and outer PF boundaries (as the PF is well defined by its boundary). Furthermore, the experts will examine anonymously two different types of images (normal and diabetic) and try to define the presence of any kind of lesion and abnormalities.

3.5.5 Statistical analysis

For the visual evaluation scoring results, different statistical tests were performed to demonstrate the significant positive relationship (Inter-operator variability) between Expert 1 and Expert 2, including multiple regression analysis and paired t-test statistics . The alpha value for statistical significance was set at 0.025 based on a Bonferroni correction. The Shapiro-Wilk test in Prism software has demonstrated that all gathered data were normally distributed. All the statistical analyses were computed using GraphPad

Prism Software version 7.01 (GraphPad Software, CA, USA).

3.6 Experimental results and discussion

Real US images of the human foot (a total of 286 different PF US images) were used to compare the presented methods. Owing to the absence of a reference image in the US domain, two different PF US images dataset were used to better asses the despeckling perfermance of different filters, including, (1) original PF US images without any modification and assuming they were already affected by the speckle noise during the process of acquisition and transmission, and (2) modified PF US images using a simulation of speckle noise-corrupted datasets; assuming they were acquired and scanned as speckle noise-free images according to the general medical scan protocol citeCrofts2014 to obtain the best possible PF US images (i.e. ground truth images). In order to get a more realistic simulation, MatLab software was used to add the multiplicative noise (with variance $S = 0.05$) to the PF US images using equation (2.2) as described in Section 2.2. In this section, we present the results of 7 groups and 16 despeckling methods, as described earlier in Section 3.4 and summarized in Table 3.1. With the aim of selecting the best-performing speckle-noise reduction approaches and to show how best to get speckle-free images in terms of high global image quality improvement and PF features (edges) preservation. Different qualitative and quantitative evaluations among these approaches will be presented in this section, including 11 quantitative image quality measures (presented in Subsection 3.5.1), feature ranking and selection analysis, and visual scoring evaluation carried out by two medical experts. The disagreement and the agreement (Inter-operator variability) between the two medical experts is assessed using linear regression and paired t-test statistical analysis for original unmodified images and simulated speckled images, respectively. So, It should be noted here that the best filtering methods are the ones with the highest image quality measures, the highest ranking order scores, the highest visual scoring demonstrating a significant positive pairing between the two experts, and the ones which are best at preserving edges and useful information in PF high texture areas. The results of this study are visually presented in Figures 3.2, 3.3, 3.4 and 3.5, and tabulated in Tables 3.3, 3.4, 3.6 and 3.5 for the original / speckle simulated images and despeckeled images, respectively.

3.6.1 Visual representation of different despeckling filters

Figures 3.2 and 3.3 show the real abnormal US images (acquired directly from US machines) with 256 grey levels and a size of 512 × 512 pixels processed using different despeckling methods for visual inspection. The visual comparison of despeckled PF US images carried out by the two experts, indicates that the best despeckling results were obtained by filters DT-CWT_H and DT-CWT_S. The filters DT-CWT_T, DT-CWT_B, DLWFDW and HybridMedian showed fairly good visual results. Other filters such as PMAD, SRAD, DPAD, DWT and DT-CWT showed fair improvement at the edges but over-smoothed the outside edges and as a consequence some diagnostic information may be lost. The remaining filters such as Median, Homog, Geom, MeanV and Wiener showed poorer visual inspection results due to the blurring effect that affects the global image quality, the PF region definition and the PF edge preservation.

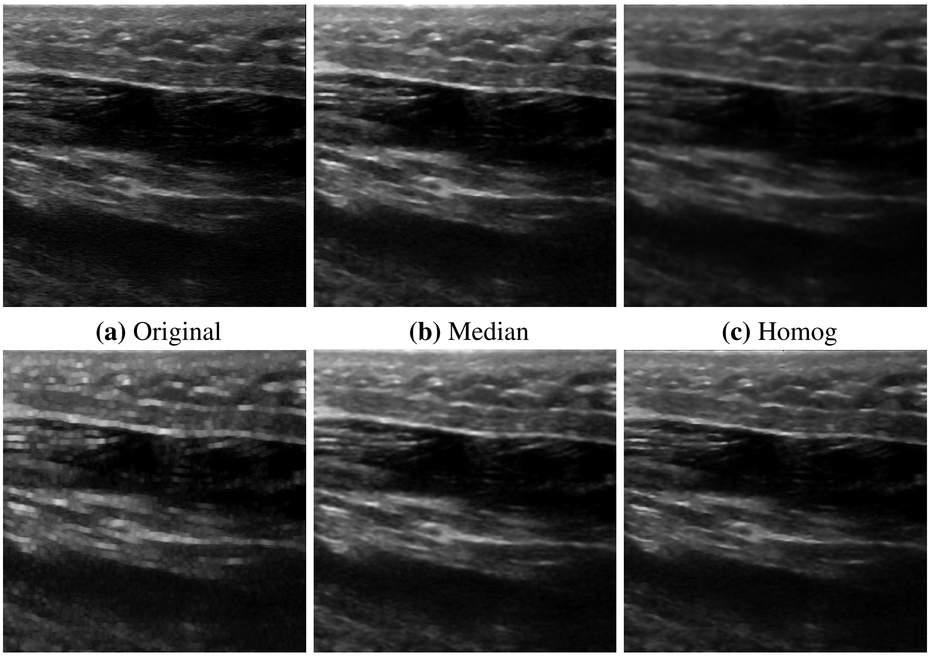

(a) Original **(b)** Median **(c)** Homog
(d) Geom **(e)** MeanV **(f)** Wiener

Figure 3.2: Comparison between different despeckling methods for real abnormal (unmodified) PF US images, (a) Original US image of the midfoot part of the plantar fascia region; (b) Median filter; (c) Homogeneity; (d) Geometric; (e) Mean variance; and (f) Wiener filter.

Figures 3.4 and 3.5 show the second results of the normal PF US images (corrupted

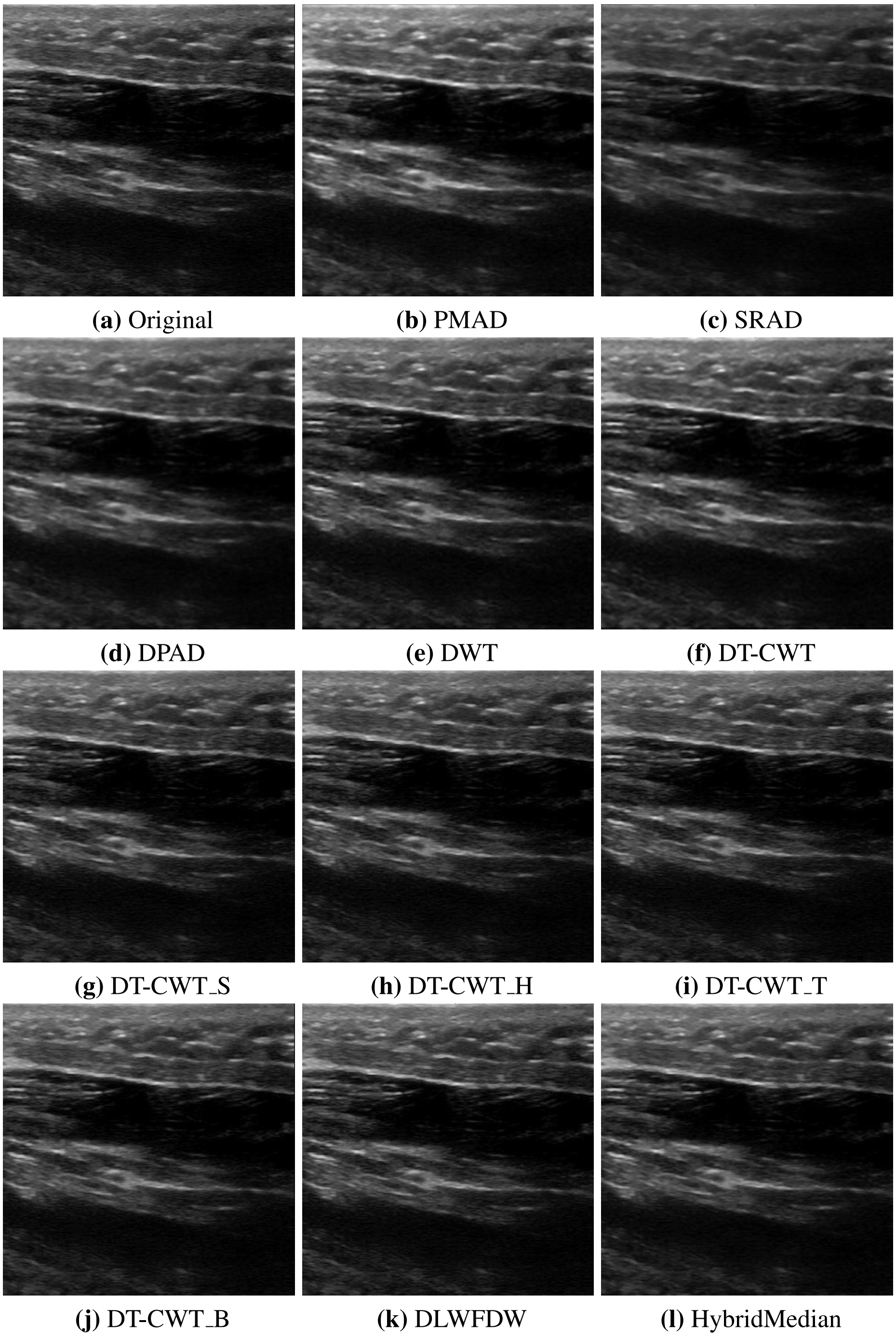


Figure 3.3: Comparison between different despeckling methods for abnormal US images of the midfoot part of the plantar fascia area, (a) Original PF image; (b) PMAD; (c) SRAD; (d) DPAD; (e) DWT; (f) DT-CWT; (g) DT-CWT_S; (h) DT-CWT_H; (i) DT-CWT_T; (j) DT-CWT_B; (k) DLWFDW; and (l) HybridMedian filter

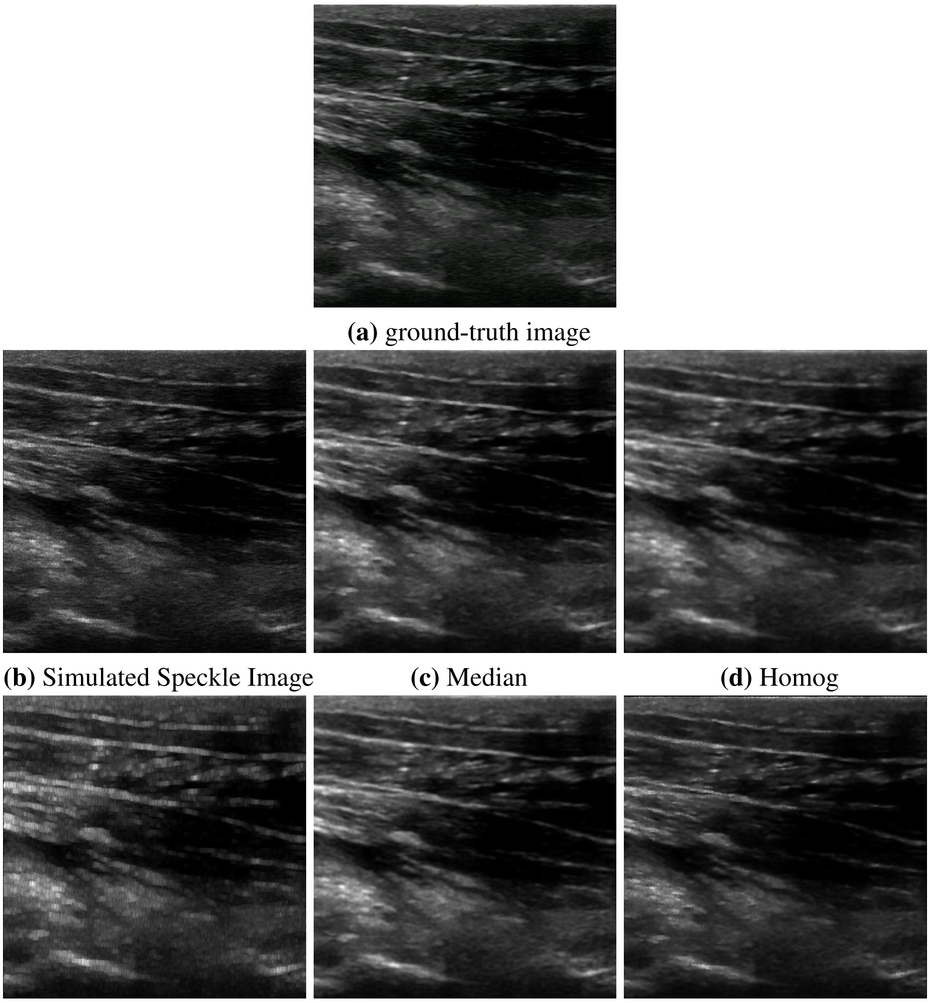
by simulated speckle noise with a variance $\sigma^2 = 0.05$) despeckled using different despeckling filters. The visual inspection of the filtered speckled US images carried out by the two experts, revealed that filters DT-CWT_H, DT-CWT_S and DT-CWT_T gave best visual results in terms of global image quality improvement, PF region definition and edge preservation criteria. Other filters such as DLWFDW, DPAD, Wiener, SRAD, DT-CWT, HybridMedian, PMAD and MeanV also gave good visual results but Wiener, HybridMedian and DLWFDW failed to reduce simulated speckle noise from the edges and thus affecting PF region definition. Finally, filters Geom, Homog, Median and DT-CWT_B showed slightly over-smoothing and a blurring impact leading to poorer visual results.

3.6.2 Quantitative image quality evaluation metrics

Table 3.3 compares the quantitative performance metrics of various existing speckle reduction filters presented in this study for original (unmodified) normal and abnormal PF US images. The bold means and standard deviation numbers indicate the best calculated image quality metrics for various despeckling methods.

Table 3.3: Quantitative performance evaluation metrics of various speckle reduction filters (mean ± STD) for normal and abnormal PF US images (a total of 286 images).

Normal Images											
Filters	**MSE**[1]	**SNR**	**RMSE**	**PSNR**	**ERR3**	**ERR4**	**UQI**	**SSIM**	**AD**	**CoC**	**EPI**
Median[2]	13.54 ± 3.71	26.41 ± 1.84	3.64 ± 0.51	37.66 ± 1.17	5.25 ± 0.84	8.05 ± 2.13	0.8 ± 0.04	0.92 ± 0.01	2.31 ± 0.4	0.9929 ± 0.003	0.9876 ± 0.010
Homog	42.17 ± 23.87	22.27 ± 1.54	6.18 ± 2.00	33.55 ± 2.13	8.34 ± 2.40	10.91 ± 2.77	0.67 ± 0.09	0.85 ± 0.06	4.28 ± 1.55	0.9805 ± 0.008	0.9875 ± 0.008
Geom	266.09 ± 132.55	14.65 ± 1.14	15.8 ± 4.08	25.13 ± 1.46	22.46 ± 5.67	28.98 ± 7.16	0.43 ± 0.03	0.69 ± 0.05	8.96 ± 2.48	0.9249 ± 0.023	0.9868 ± 0.009
MeanV	26.26 ± 9.72	23.67 ± 1.38	5.03 ± 0.97	34.94 ± 1.19	7.16 ± 1.45	9.98 ± 2.20	0.75 ± 0.04	0.89 ± 0.02	3.42 ± 0.76	0.9868 ± 0.004	0.9855 ± 0.011
Wiener	12.74 ± 4.74	26.86 ± 1.41	3.50 ± 0.69	38.09 ± 1.3	4.20 ± 0.76	4.79 ± 0.84	0.77 ± 0.04	0.91 ± 0.02	2.64 ± 0.63	0.9935 ± 0.002	0.9911 ± 0.006
PMAD	21.03 ± 9.47	24.82 ± 1.17	4.46 ± 1.07	36.09 ± 1.48	8.16 ± 2.13	12.82 ± 3.3	0.89 ± 0.02	0.95 ± 0.01	2.46 ± 0.59	0.9896 ± 0.004	0.9864 ± 0.008
SRAD	10.43 ± 4.02	27.73 ± 1.25	3.17 ± 0.64	38.97 ± 1.18	4.05 ± 0.72	5.21 ± 1.62	0.9 ± 0.02	0.95 ± 0.01	2.28 ± 0.58	0.995 ± 0.001	0.9955 ± 0.004
DPAD	14.4 ± 5.12	26.26 ± 1.53	3.73 ± 0.68	37.51 ± 1.17	4.69 ± 0.81	5.69 ± 0.98	0.83 ± 0.03	0.93 ± 0.01	2.70 ± 0.60	0.9929 ± 0.002	0.9918 ± 0.007
DWT	15.1 ± 3.48	25.92 ± 1.96	3.86 ± 0.46	37.14 ± 1.22	4.66 ± 0.45	5.42 ± 0.47	0.78 ± 0.04	0.91 ± 0.01	2.91 ± 0.5	0.9917 ± 0.003	0.9872 ± 0.010
DT-CWT	15.68 ± 3.81	25.76 ± 1.94	3.93 ± 0.49	36.98 ± 1.22	4.85 ± 0.47	5.84 ± 0.50	0.78 ± 0.04	0.91 ± 0.02	2.93 ± 0.53	0.9915 ± 0.003	0.9879 ± 0.010
DT-CWT_S	1.22 ± 0.08	36.8 ± 2.78	1.11 ± 0.03	47.93 ± 1.71	1.19 ± 0.06	1.30 ± 0.09	**0.94 ± 0.08**	**0.99 ± 0.01**	1.01 ± 0.01	**0.9999 ± 0.001**	0.9992 ± 0.001
DT-CWT_H	**1.12 ± 0.05**	**37.18 ± 2.82**	**1.06 ± 0.02**	**48.31 ± 1.74**	**1.11 ± 0.04**	**1.18 ± 0.06**	**0.94 ± 0.08**	**0.99 ± 0.01**	1.01 ± 0.01	**0.9999 ± 0.001**	**0.9995 ± 0.001**
DT-CWT_T	**1.13 ± 0.05**	**37.15 ± 2.82**	**1.06 ± 0.02**	**48.28 ± 1.73**	**1.12 ± 0.04**	**1.19 ± 0.07**	**0.94 ± 0.08**	**0.99 ± 0.01**	1.01 ± 0.01	**0.9999 ± 0.001**	**0.9995 ± 0.001**
DT-CWT_B	1.37 ± 0.23	36.32 ± 2.61	1.17 ± 0.09	47.47 ± 1.65	1.32 ± 0.15	1.55 ± 0.38	**0.94 ± 0.08**	**0.99 ± 0.01**	**1.00 ± 0.05**	**0.9997 ± 0.001**	**0.9996 ± 0.001**
DLWFDW	**0.29 ± 0.07**	**43.11 ± 2.04**	**0.53 ± 0.07**	**54.31 ± 1.39**	**0.69 ± 0.07**	**0.82 ± 0.08**	**0.99 ± 0.01**	**0.99 ± 0.01**	**0.27 ± 0.06**	**0.9998 ± 0.001**	0.9989 ± 0.001
HybridMedian	2.42 ± 0.57	33.86 ± 2.00	1.55 ± 0.19	45.08 ± 1.26	2.20 ± 0.31	3.02 ± 1.04	0.96 ± 0.01	0.98 ± 0.01	**0.8 ± 0.13**	0.9987 ± 0.001	0.9962 ± 0.003



(a) ground-truth image

(b) Simulated Speckle Image

(c) Median

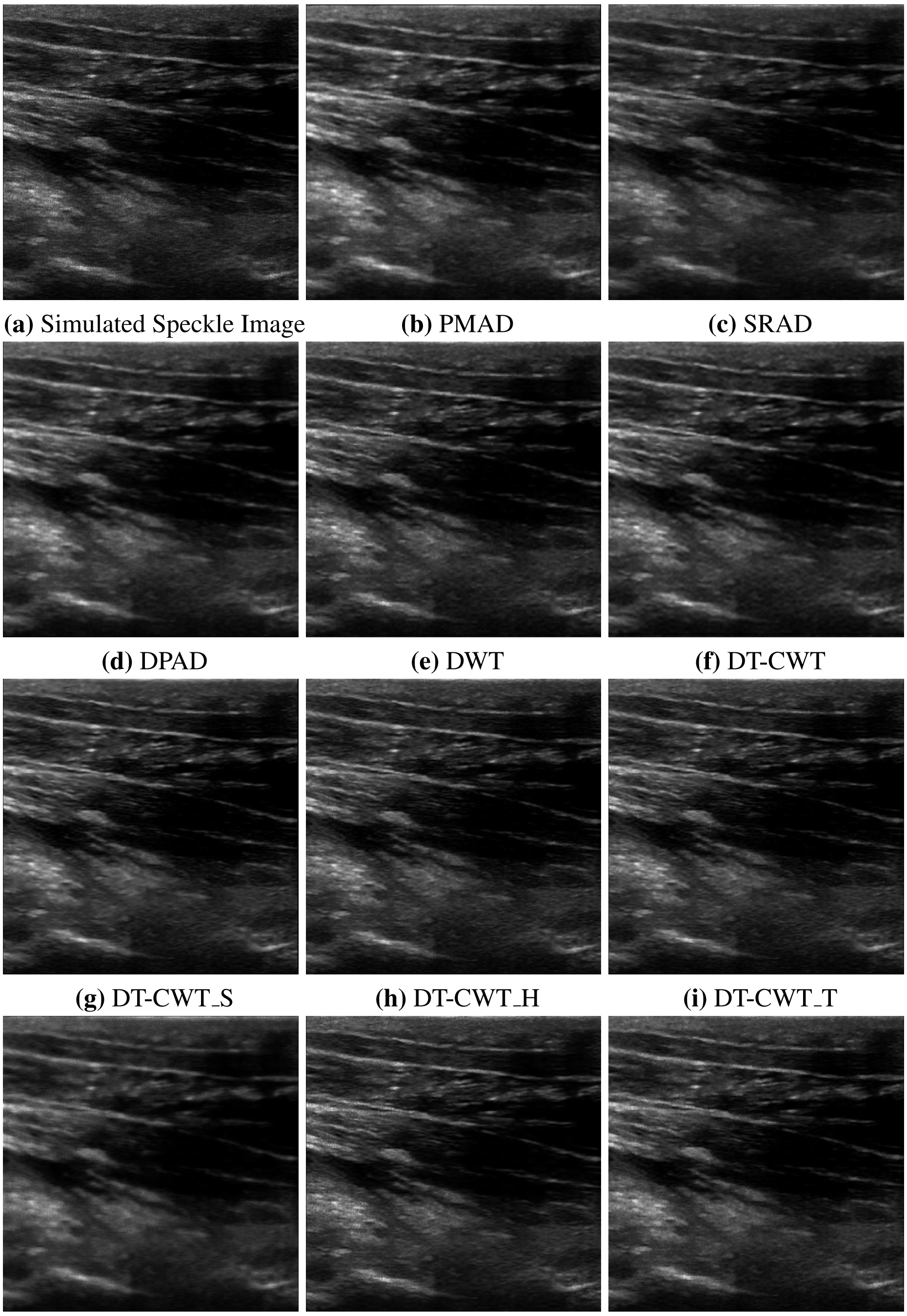
(d) Homog

(e) Geom

(f) MeanV

(g) Wiener

Figure 3.4: Comparison between different despeckling methods for real normal PF US images (corrupted by simulated speckle noise with variance=5), (a) ground-truth PF image, (b) simulated speckle image of the midfoot part of the plantar fascia region; (c) Median; (d) Homogeneity; (e) Geometric; (f) Mean variance; and (g) Wiener filter.



(a) Simulated Speckle Image **(b)** PMAD **(c)** SRAD

(d) DPAD **(e)** DWT **(f)** DT-CWT

(g) DT-CWT_S **(h)** DT-CWT_H **(i)** DT-CWT_T

(j) DT-CWT_B **(k)** DLWFDW **(l)** HybridMedian

Figure 3.5: Comparison between different despeckling methods for real normal PF US images (corrupted by simulated speckle noise with variance=0.05), (a) Original noisy US image of the midfoot part of the plantar fascia region; (b) PMAD; (c) SRAD; (d) DPAD; (e) DWT; (f) DT-CWT; (g) DT-CWT_S; (h) DT-CWT_H; (i) DT-CWT_T; (j) DT-CWT_B; (k) DLWFDW; and (l) HybridMedian filter

Abnormal Images

Filters	MSE[1]	SNR	RMSE	PSNR	ERR3	ERR4	UQI	SSIM	AD	CoC	EPI
Median	19.981 ± 10.192	24.7038 ± 2.495	4.3648 ± 0.978	37.3322 ± 1.552	6.7317 ± 2.673	10.4993 ± 4.492	0.767 ± 0.045	0.9056 ± 0.014	2.5955 ± 0.453	0.9899 ± 0.007	0.9782 ± 0.026
Homog	53.4819 ± 25.054	20.6885 ± 2.109	7.0634 ± 1.922	33.3665 ± 2.708	9.7082 ± 2.485	12.6161 ± 3.250	0.6137 ± 0.088	0.8277 ± 0.057	4.7602 ± 1.503	0.9751 ± 0.010	0.9780 ± 0.023
Geom	311.736 ± 134.656	13.6775 ± 0.857	17.2401 ± 3.864	25.4353 ± 1.747	25.0617 ± 4.923	32.8658 ± 6.034	0.4165 ± 0.028	0.6812 ± 0.053	9.6037 ± 2.526	0.9147 ± 0.021	0.9818 ± 0.011
MeanV	34.9789 ± 11.341	22.1121 ± 1.788	5.8321 ± 0.997	34.7614 ± 1.259	8.7419 ± 1.798	12.6574 ± 2.913	0.7122 ± 0.04	0.8738 ± 0.021	3.8222 ± 0.796	0.9830 ± 0.006	0.9740 ± 0.028
Wiener	15.559 ± 5.462	25.7397 ± 1.146	3.8786 ± 0.728	38.3344 ± 1.5	4.6367 ± 0.807	5.2863 ± 0.882	0.7365 ± 0.038	0.8931 ± 0.021	2.9326 ± 0.658	0.9926 ± 0.002	0.9878 ± 0.006
PMAD	28.1254 ± 10.363	23.1277 ± 1.847	5.2114 ± 0.997	35.771 ± 1.388	9.9045 ± 2.065	15.8039 ± 3.292	0.8672 ± 0.02	0.9424 ± 0.009	2.7637 ± 0.604	0.9864 ± 0.005	0.9759 ± 0.024
SRAD	12.5237 ± 4.633	26.6917 ± 1.124	3.4753 ± 0.677	39.2973 ± 1.501	4.5406 ± 0.818	5.8777 ± 1.727	0.8927 ± 0.011	0.9458 ± 0.013	2.457 ± 0.622	0.9943 ± 0.001	0.9936 ± 0.005
DPAD	18.9909 ± 6.524	24.8307 ± 1.774	4.2891 ± 0.782	37.449 ± 1.37	5.7014 ± 1.459	7.5057 ± 2.467	0.8113 ± 0.026	0.9175 ± 0.014	3.0048 ± 0.64	0.9910 ± 0.003	0.9855 ± 0.017
DWT	17.2821 ± 3.814	25.1030 ± 1.743	4.1302 ± 0.480	37.6855 ± 1.082	5.0377 ± 0.457	5.9368 ± 0.458	0.7607 ± 0.048	0.8999 ± 0.013	3.1003 ± 0.516	0.9913 ± 0.002	0.9830 ± 0.011
DT-CWT	19.1846 ± 4.247	24.6416 ± 1.766	4.3512 ± 0.509	37.2341 ± 1.069	5.4267 ± 0.476	6.6332 ± 0.544	0.7527 ± 0.042	0.8929 ± 0.016	3.2273 ± 0.564	0.9904 ± 0.002	0.9838 ± 0.011
DT-CWT_S	1.1843 ± 0.057	36.719 ± 2.558	1.088 ± 0.026	49.214 ± 1.171	1.1652 ± 0.046	1.2609 ± 0.067	0.944 ± 0.066	0.9875 ± 0.012	1.0066 ± 0.006	**0.9999 ± 0.001**	**0.9993 ± 0.001**
DT-CWT_H	**1.1004 ± 0.03**	**37.0366 ± 2.613**	**1.0489 ± 0.014**	**49.530 ± 1.200**	**1.0937 ± 0.026**	**1.1542 ± 0.042**	**0.9448 ± 0.066**	**0.9881± 0.012**	**1.0037 ± 0.003**	**0.9999 ± 0.001**	**0.9996 ± 0.001**
DT-CWT_T	**1.1071 ± 0.033**	**37.0102 ± 2.609**	**1.0521 ± 0.015**	**49.5037 ± 1.197**	**1.0996 ± 0.028**	**1.1634 ± 0.044**	**0.9447 ± 0.066**	**0.9881 ± 0.012**	**1.0042 ± 0.003**	**0.9999 ± 0.001**	**0.9996 ± 0.001**
DT-CWT_B	1.5624 ± 0.806	35.7686 ± 2.924	1.227 ± 0.242	48.2864 ± 1.575	1.5029 ± 0.758	1.8483 ± 1.391	0.9427 ± 0.066	0.9871 ± 0.011	1.0131 ± 0.058	0.9996 ± 0.001	**0.9993 ± 0.001**
DLWFDW	**0.2898 ± 0.079**	**42.9579 ± 1.867**	**0.5327 ± 0.079**	**55.5181 ± 1.342**	**0.699 ± 0.082**	**0.8301 ± 0.089**	**0.9878 ± 0.009**	**0.9978 ± 0.001**	**0.2606 ± 0.065**	**0.9999 ± 0.001**	0.9989 ± 0.001
HybridMedian	2.9199 ± 0.971	32.9759 ± 2.013	1.6840 ± 0.294	45.5566 ± 1.413	2.7735 ± 0.949	4.9210 ± 2.422	0.9592 ± 0.010	0.9832 ± 0.003	0.8582 ± 0.165	0.9986 ± 0.001	0.9950 ± 0.004

By examining Table 3.3, it is evident that the best values were achieved by DLWFDW followed by DT-CWT_H, DT-CWT_T, DT-CWT_S, DT-CWT_B and HybridMedian with higher PSNR and SNR and lower RMSE, MSE, ERR3 and ERR4 with the computed values of UQI and SSIM being close to unity for similar US images and effective despeckling for both (unmodified) normal and abnormal PF US images. But on the other hand, the best values for AD were obtained for DLWFDW followed by HybridMedian and DT-CWT_B for normal images, and DLWFDW, DT-CWT_H, DT-CWT_T for abnormal images. The filters DT-CWT_S, DT-CWT_H, DT-CWT_T, DLWFDW, DT-CWT_B and HybridMedian gave best performance values in terms of higher CoC. However, in terms of high edge preserving index, filters DT-CWT_B, DT-CWT_S and DT-CWT_H were superior in comparison to other filtering methods for normal images. For abnormal images, the best values for CoC were obtained for filters DLWFDW, DT-CWT_H, DT-CWT_T, DT-CWT_S and DT-CWT_B. Moreover, in terms of higher edge preserving index EPI values, filters DT-CWT_H and DT-CWT_T gave the best performance, followed by DT-CWT_S and DT-CWT_B filters.

Based on the second filtering analysis using a simulated speckle noise (with noise variance $\sigma^2 = 0.05$) as presented in Table 3.4, it is observed that filter DT-CWT_S gave best image quality performance results followed by DT-CWT_T, DT-CWT_H and DPAD in terms of: lower MSE, RMSE, ERR3, ERR4 and AD; higher SNR, PSNR; and the computed values of UQI and SSIM being close to unity for both affected normal and abnormal PF US images. But in terms of high edge preserving index, filter DT-CWT_S gave best results followed by DPAD and DT-CWT filters.

From the overall results shown in Table 3.3 regarding the first set of images (unmodified images), it is apparent that the DT-CWT filtering group using the BayesShrink concept and different thresholding rules (wiener followed by hard, trimmed, soft and bivariate) and HybridMedian exhibit the best performance in terms of most calculated image quality metrics for both normal and abnormal PF US images. Filter DLWFDW (using hybrid model) gave very good performance in terms of lower MSE, RMSE, ERR3, ERR4 and AD; higher SNR, PSNR, UQI and SSIM. But in terms of high edge preserving index, filters DT-CWT_B and DT-CWT_T performed better for normal and abnormal PF US images, respectively.

When comparing anisotropic diffusion-based group that uses the anisotropic diffusion concept with other remaining despeckling filters such as Median, MeanV, Wiener, Homog, Geom, it is very clear from Table 3.3 that SRAD filter gave best results in terms of all calculated image quality metrics excluding ERR4 for normal and abnormal PF US images, respectively. For ERR4, Wiener filter performed better with values of 4.79 and 5.29 for normal and abnormal images, respectively. Regarding the second results as shown in Table 3.3, DPAD filters overcome other filters such as SRAD, PMAD, Median, MeanV, Wiener, Homog and Geom in terms of all computed measures for both normal and abnormal images. These results indicate that the SRAD and DPAD filters performed better when applied to high speckeled and low speckled ultrasound images, respectively. However the SRAD and DPAD filter need to be improved in terms of image quality assessment parameters and time complexity. This is because, these filters are non-linear techniques which usually require many iteration steps compared with some other filters. In this study Homog, Geom, MeanV and PMAD filters performed poorer when compared to other filtering methods as they failed to reduce the speckle noise in some areas of the images and to improve the global image quality.

By examining the second results as presented in Table 3.4 concerning the second set of images (images affected by simulated speckle noise with a variance of $\sigma^2 = 0.05$), it is also very clear that the DT-CWT filtering group using the BayesShrink concept and thresholding rules such as soft followed by trimmed and hard present the best performance in terms of most calculated image quality metrics such as MSE, RMSE, ERR3, ERR4, AD, SNR, PSNR, UQI and SSIM for both normal and abnormal PF US images.

But in respect of edge preserving index EPI, filters DT-CWT_S, DPAD and DT-CWT performed better.

It is also found, that there is a significant improvement of the BayesShrink thresholding approach (DLWFDW, DT-CWT_H, DT-CWT_T and DT-CWT_S) in terms of image quality assessment metrics over the SureShrink thresholding scheme (DWT and DT-CWT). This is because the BayesShrink is an adaptive and sub-band dependent thresholding method that minimizes the Bayesian risk. In contrast the SureShrink thresholding scheme relies on the selection of its universal threshold, which can be very large because it is directly related to the image size value (N) for a typical image of size 512×512, and this can yield overly smoothed images and reduce too many useful coefficients.

Furthermore, some previous studies had successfully introduced wavelet based despeckling proposed earlier by Donoho (1995) in different images such as SAR (Lee, 1981a; Medeiros et al., 2002) and ultrasound images (Achim et al., 2001; Zhong and Cherkassky, 2000) with a satisfactory results. Similarly, this study has proven the use of DT-CWT filtering integrating the BayesShrink concept and some thresholding rules to improve the image quality of the PF US images in terms of high speckle reduction and edge preservation.

Table 3.4: Quantitative performance evaluation metrics of various speckle reduction filters for normal and abnormal PF US images (a total of 286 images) with simulated speckle noise variance $\sigma^2 = 0.05$ (mean ± STD).

	Normal Images										
Filters	**MSE**	**SNR**	**RMSE**	**PSNR**	**ERR3**	**ERR4**	**UQI**	**SSIM**	**AD**	**CoC**	**EPI**
Original Image	62.0795 ± 33.51	20.4981 ± 0.033	7.5586 ± 2.229	31.6614 ± 1.814	9.7327 ± 2.609	11.719 ± 2.969	0.768 ± 0.074	0.8112 ± 0.071	5.1061 ± 1.864	0.972 ± 0.01	0.9055 ± 0.017
Median	27.4995 ± 10.892	23.4939 ± 1.123	5.1353 ± 1.064	34.785 ± 1.181	6.8358 ± 1.3	9.2988 ± 2.128	0.7291 ± 0.036	0.8789 ± 0.027	3.5978 ± 0.911	0.9864 ± 0.004	0.9826 ± 0.011
Homog	47.5456 ± 24.522	21.4784 ± 1.013	6.6331 ± 1.887	32.7673 ± 1.596	8.9224 ± 2.331	11.6092 ± 2.728	0.6347 ± 0.075	0.8405 ± 0.05	4.5767 ± 1.468	0.9774 ± 0.008	0.9838 ± 0.016
Geom	416.062 ± 207.191	13.0673 ± 0.8	19.7298 ± 5.187	23.2184 ± 1.458	26.4589 ± 6.604	33.2771 ± 8.048	0.3511 ± 0.049	0.5921 ± 0.078	13.2417 ± 4.032	0.9131 ± 0.026	0.9479 ± 0.017
MeanV	31.7542 ± 12.612	22.8987 ± 1.179	5.5169 ± 1.15	34.1652 ± 1.21	7.6401 ± 1.581	10.3858 ± 2.257	0.7223 ± 0.035	0.8781 ± 0.024	3.7853 ± 0.921	0.984 ± 0.005	0.9844 ± 0.011
Wiener	29.8779 ± 12.351	23.2679 ± 0.904	5.3383 ± 1.177	34.4818 ± 1.247	6.9644 ± 1.49	8.6728 ± 1.836	0.7263 ± 0.034	0.8749 ± 0.028	3.7499 ± 0.953	0.9852 ± 0.004	0.9647 ± 0.017
PMAD	29.5005 ± 13.79	23.402 ± 0.879	5.2693 ± 1.32	34.661 ± 1.473	8.619 ± 2.192	13.0285 ± 3.304	0.8328 ± 0.02	0.920 ± 0.02	3.2095 ± 0.898	0.9855 ± 0.005	0.9852 ± 0.008
SRAD	23.2687 ± 10.762	24.4419 ± 0.667	4.6844 ± 1.153	35.6726 ± 1.389	6.1366 ± 1.346	7.9554 ± 1.951	0.8285 ± 0.037	0.9034 ± 0.03	3.2527 ± 0.974	0.9886 ± 0.004	0.9748 ± 0.008
DPAD	20.7835 ± 8.448	24.773 ± 1.132	4.4604 ± 0.944	36.0161 ± 1.206	5.649 ± 1.123	6.8863 ± 1.315	0.796 ± 0.023	0.9092 ± 0.019	3.2021 ± 0.807	0.9897 ± 0.003	**0.9911 ± 0.007**
DWT	27.1169 ± 11.19	23.6503 ± 1.095	5.0941 ± 1.082	34.8607 ± 1.237	6.4217 ± 1.33	7.7622 ± 1.601	0.7383 ± 0.033	0.8776 ± 0.025	3.6849 ± 0.904	0.9864 ± 0.004	0.9818 ± 0.009
DT-CWT	23.0286 ± 7.944	24.2345 ± 1.403	4.7248 ± 0.841	35.4543 ± 1.154	5.9183 ± 0.946	7.1804 ± 1.078	0.7505 ± 0.034	0.889 ± 0.021	3.4513 ± 0.772	0.988 ± 0.004	**0.9873 ± 0.01**
DT-CWT_S	**14.3138 ± 6.248**	**26.5195 ± 0.828**	**3.6875 ± 0.848**	**37.7104 ± 1.323**	**4.7082 ± 1.011**	**5.7698 ± 1.174**	**0.834 ± 0.058**	**0.9285 ± 0.013**	**2.6567 ± 0.65**	**0.9931 ± 0.002**	**0.9917 ± 0.007**
DT-CWT_H	**17.7969 ± 7.659**	**25.5564 ± 1.208**	**4.12 ± 0.909**	**36.7189 ± 1.344**	**5.3786 ± 1.133**	**6.6769 ± 1.365**	**0.8358 ± 0.065**	**0.9214 ± 0.015**	**2.8922 ± 0.679**	**0.9914 ± 0.002**	0.9829 ± 0.015
DT-CWT_T	**16.613 ± 7.039**	**25.8422 ± 1.188**	**3.983 ± 0.867**	**37.0082 ± 1.321**	**5.1747 ± 1.07**	**6.4034 ± 1.279**	**0.839 ± 0.065**	**0.9249 ± 0.014**	**2.8128 ± 0.653**	**0.992 ± 0.002**	0.9854 ± 0.014
DT-CWT_B	40.486 ± 19.499	22.0403 ± 0.908	6.1665 ± 1.572	33.3034 ± 1.442	8.1465 ± 1.984	10.3686 ± 2.432	0.6659 ± 0.033	0.8484 ± 0.03	4.3269 ± 1.173	0.9805 ± 0.006	0.9792 ± 0.011
DLWFDW	25.9878 ± 10.958	23.9076 ± 1.125	4.9744 ± 1.117	35.1045 ± 1.354	6.6085 ± 1.43	8.2813 ± 1.759	0.775 ± 0.05	0.8975 ± 0.025	3.3868 ± 0.885	0.9875 ± 0.002	0.9693 ± 0.021
HybridMedian	27.0793 ± 13.456	23.8669 ± 0.429	5.0291 ± 1.34	35.1094 ± 1.539	6.5586 ± 1.603	8.1187 ± 1.85	0.8219 ± 0.039	0.8931 ± 0.036	3.4233 ± 1.102	0.987 ± 0.004	0.9699 ± 0.009

Abnormal Images

Filters	**MSE**	**SNR**	**RMSE**	**PSNR**	**ERR3**	**ERR4**	**UQI**	**SSIM**	**AD**	**CoC**	**EPI**
Original Image	60.0044 ± 35.490	20.4243 ± 0.286	7.4185 ± 2.261	32.9448 ± 2.393	9.7267 ± 2.521	11.9075 ± 2.724	0.7942 ± 0.064	0.832 ± 0.07	4.9266 ± 1.926	0.9742 ± 0.009	0.9053 ± 0.018
Median	34.5005 ± 13.672	22.2546 ± 1.821	5.7565 ± 1.184	34.9302 ± 1.519	8.1891 ± 2.529	11.6345 ± 4.271	0.704 ± 0.03	0.8659 ± 0.028	3.8445 ± 0.941	0.9835 ± 0.007	0.9724 ± 0.027
Homog	514.5450 ± 196.049	19.6838 ± 4.524	11.7622 ± 19.671	31.947 ± 5.464	15.4829 ± 24.454	19.1559 ± 27.583	0.5937 ± 0.135	0.7909 ± 0.163	7.7392 ± 12.281	0.9457 ± 0.122	0.9451 ± 0.151
Geom	462.7996 ± 217.490	12.3604 ± 0.625	20.9144 ± 5.11	23.7997 ± 1.905	28.8312 ± 5.945	37.0224 ± 6.805	0.3493 ± 0.038	0.5904 ± 0.081	13.6462 ± 4.187	0.904 ± 0.024	0.9433 ± 0.018
MeanV	40.2054 ± 13.562	21.5327 ± 1.623	6.2452 ± 1.112	34.1787 ± 1.312	9.1482 ± 1.815	12.9993 ± 2.886	0.6899 ± 0.034	0.8605 ± 0.026	4.1303 ± 0.934	0.9804 ± 0.006	0.9729 ± 0.028
Wiener	32.7031 ± 12.970	22.5901 ± 0.903	5.6036 ± 1.158	35.165 ± 1.639	7.3605 ± 1.341	9.2831 ± 1.52	0.7 ± 0.031	0.8644 ± 0.03	3.9423 ± 0.981	0.9843 ± 0.003	0.9593 ± 0.017
PMAD	36.3012 ± 13.611	22.0426 ± 1.551	5.915 ± 1.163	34.6821 ± 1.431	10.2936 ± 2.061	15.9933 ± 3.27	0.8221 ± 0.014	0.9157 ± 0.02	3.418 ± 0.881	0.9824 ± 0.005	0.9745 ± 0.025
SRAD	25.2374 ± 11.600	23.8042 ± 0.738	4.8924 ± 1.157	36.3995 ± 1.796	6.6955 ± 1.167	9.321 ± 1.982	0.8327 ± 0.027	0.9047 ± 0.032	3.3265 ± 1.013	0.9882 ± 0.003	0.9721 ± 0.01
DPAD	25.0943 ± 9.156	23.6608 ± 1.480	4.9207 ± 0.952	36.2752 ± 1.447	6.5381 ± 1.495	8.4876 ± 2.421	0.7812 ± 0.02	0.9005 ± 0.02	3.4296 ± 0.824	0.988 ± 0.003	**0.9848 ± 0.017**
DWT	29.3738 ± 11.363	23.0210 ± 1.031	5.3202 ± 1.049	35.5938 ± 1.527	6.8095 ± 1.207	8.4122 ± 1.368	0.7259 ± 0.035	0.8739 ± 0.025	3.8189 ± 0.901	0.9859 ± 0.003	0.9763 ± 0.01
DT-CWT	26.4269 ± 8.077	23.3532 ± 1.363	5.0792 ± 0.804	35.9394 ± 1.268	6.4967 ± 0.833	8.1311 ± 0.942	0.7278 ± 0.035	0.8789 ± 0.021	3.6733 ± 0.776	0.987 ± 0.002	**0.9826 ± 0.011**
DT-CWT_S	**14.8825 ± 6.572**	**26.0740 ± 0.995**	**3.770 ± 0.830**	**38.6207 ± 1.694**	**4.9371 ± 0.905**	**6.2997 ± 0.993**	**0.8474 ± 0.049**	**0.9321 ± 0.015**	**2.672 ± 0.671**	**0.9932 ± 0.002**	**0.9893 ± 0.007**
DT-CWT_H	**20.0899 ± 7.796**	**24.7232 ± 1.284**	**4.4007 ± 0.863**	**37.238 ± 1.676**	**5.8741 ± 0.954**	**7.4577 ± 1.019**	**0.8392 ± 0.054**	**0.9196 ± 0.015**	**3.0245 ± 0.696**	**0.9907 ± 0.002**	0.978 ± 0.014
DT-CWT_T	**18.6612 ± 7.223**	**25.0390 ± 1.283**	**4.2416 ± 0.83**	**37.5574 ± 1.662**	**5.6431 ± 0.911**	**7.1614 ± 0.967**	**0.8436 ± 0.054**	**0.9238 ± 0.014**	**2.9316 ± 0.672**	**0.9913 ± 0.002**	0.981 ± 0.013
DT-CWT_B	45.6177 ± 21.311	21.2529 ± 1.269	6.5595 ± 1.633	33.8954 ± 1.937	9.1624 ± 2.467	12.0529 ± 3.808	0.69 ± 0.031	0.8529 ± 0.036	4.3998 ± 1.257	0.9788 ± 0.007	0.973 ± 0.016
DLWFDW	26.2374 ± 9.324	23.4962 ± 1.163	5.0386 ± 0.935	36.0502 ± 1.524	6.8276 ± 1.015	8.7202 ± 1.123	0.7914 ± 0.035	0.9027 ± 0.025	3.3787 ± 0.855	0.9875 ± 0.002	0.9661 ± 0.02
HybridMedian	28.2649 ± 13.979	23.3746 ± 0.548	5.1558 ± 1.316	35.9845 ± 1.948	6.9672 ± 1.466	9.0926 ± 1.842	0.8317 ± 0.031	0.8977 ± 0.037	3.4342 ± 1.131	0.987 ± 0.004	0.9658 ± 0.01

[1] MSE, mean square error; SNR, signal-to-noise ratio; RMSE, randomized mean square error; PSNR, peak signal-to-noise ratio; ERR3 and ERR4, Minowski error measures; UQI, universal quality index; SSIM, structural similarity index map; AD, average difference; CoC, correlation coefficient; and EPI, edge preservation index.

[2] Bold (mean ± STD) values indicate best despeckling methods.

3.6.3 Feature ranking and feature selection analysis

Tables 3.5 and 3.6 tabulate the results of feature ranking and selection test. The two tables show the average feature weights (W) and ranking orders (R-O) for 33 features extracted both from: (1) the original PF US images (without any modification) and the filtered images using 16 different despeckling filters (Table 3.5); and (2) the corrupted PF US images with simulated speckle noise with variance of $\sigma^2 = 0.05$ and despeckling filtered images (Table 3.6).

The bold average numbers indicate improved feature weight and ranking order values after applying despeckling filters compared to the original unfiltered PF US images. A high weight and an improved ranking order show improvement after despeckling process.

Additionally, Tables 3.7 and 3.8 were generated from Tables 3.5 and 3.6, respectively, to define the best filtering methods in terms of highest selected feature weight and an improved ranking order. Both tables represent the total scores of all ranked features that are significantly different and improved after using different filtering techniques. The scores were calculated by counting the number of cases in each feature set that the ranking order (R-O) scores were significantly improved for different filtering methods. The last row represents the total scores for all improved ranking order scores in which the highest total score indicate the best despeckling filter.

For Table 3.7, best scores were achieved by DT-CWT_H, DT-CWT_T, DT-CWT_B, DWT, Median and DT-CWT_S. For Table 3.8, highest scores were obtained by DT-CWT_B, DT-CWT_S, DT-CWT_H, DT-CWT_T, HybridMedian, DT-CWT, MeanV and Median. Finally, based on the total score on each feature set row, the feature sets that demonstrated a significant change and improvement in their weights and ranking orders after despeckling process were FOS, SGLDM and FPS.

Table 3.5: Feature weights (W) and ranking order (O-R) for different feature sets (a total of 33 different features) extracted both from original and filtered images.

				Median		Homogeneity		Geometric		Local Statistics				Anisotropic Diffusion						Wavelet Based filtering												Hybrid filtering			
		Original Image		Median		Homog		Geom		MeanV		Wiener		PMAD		SRAD		DPAD		DWT		DT-CWT		DT-CWT_S		DT-CWT_H		DT-CWT_T		DT-CWT_B		DLWFDW		HybridMedian	
Feature No	Feature name	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O
First Order Statistics Histogram Features																																			
1	mean	5.519	30	5.567	29	**6.384**	29	5.486	31	5.556	31	5.657	30	5.469	33	5.493	31	5.588	31	5.508	31	5.549	30	5.555	29	**5.840**	**17**	**5.840**	**17**	**5.817**	**17**	5.553	30	5.459	31
2	st	7.331	14	7.296	14	7.161	22	7.202	14	7.121	14	7.181	15	7.290	15	**7.338**	**14**	7.232	14	7.110	15	7.123	15	7.097	15	6.835	11	6.833	11	6.886	11	**7.459**	**13**	**7.398**	**15**
3	skew	12.314	6	13.470	4	10.433	8	13.035	4	12.914	4	13.059	5	12.381	5	12.618	5	12.512	4	13.038	4	12.616	5	12.777	5	**13.754**	**3**	**13.756**	**3**	**13.968**	**3**	12.424	5	13.601	5
4	eng	5.501	32	5.585	27	**6.396**	27	**5.655**	29	**5.620**	28	5.658	29	5.542	28	5.529	28	**5.661**	28	5.531	28	5.602	27	5.568	27	5.595	22	5.595	22	5.550	21	5.519	32	5.477	29
5	ent	5.631	28	5.535	30	**6.380**	30	**5.770**	26	5.643	27	5.662	27	5.643	26	5.538	27	**5.721**	**27**	5.537	27	5.593	28	5.547	30	5.470	24	5.469	24	5.383	24	5.675	27	5.567	25
Spatial Gray Level Dependence Matrices (SGLDM)																																			
6	ang_sec_mom	6.445	17	6.415	16	6.776	25	5.761	27	6.441	17	6.419	17	6.374	17	6.422	18	6.432	17	6.502	17	6.507	17	6.534	17	**7.598**	**9**	**7.599**	**9**	**7.631**	**9**	6.312	17	6.431	17
7	cont	6.136	19	6.197	18	**9.862**	**10**	5.939	22	6.074	20	6.176	18	6.002	20	6.219	19	6.135	18	**6.222**	**18**	6.174	18	6.135	18	5.733	18	5.732	18	5.636	18	6.161	19	**6.246**	**18**
8	corr	15.568	1	11.237	6	11.567	2	10.996	7	10.469	6	13.457	4	12.681	4	15.087	1	10.456	6	10.774	5	9.759	8	10.051	8	**17.120**	**1**	**17.128**	**1**	**16.512**	**1**	15.506	1	13.751	3
9	sum_sq	7.323	15	7.293	15	7.196	21	7.161	15	7.115	15	7.182	14	**7.342**	**14**	7.300	15	7.200	15	7.126	14	7.140	14	7.113	14	7.013	10	7.011	10	7.060	10	**7.442**	**14**	**7.477**	**14**
10	inv_diff_mom	5.804	23	5.938	21	**7.150**	23	**6.627**	**16**	5.857	24	5.813	24	5.747	24	5.638	25	5.790	25	5.837	24	5.924	24	5.974	23	6.262	14	6.263	14	**6.271**	**13**	5.843	24	5.563	26
11	sum_avg	5.520	29	5.568	28	**6.384**	**28**	5.487	30	5.557	30	5.658	28	5.470	32	5.494	30	5.589	30	5.508	30	5.550	29	5.556	28	5.849	16	**5.849**	**16**	**5.827**	**16**	5.554	29	5.460	30
12	sum_var	7.395	13	7.385	13	7.224	20	7.229	13	7.162	13	7.226	13	7.389	13	7.428	13	7.245	13	7.177	13	7.198	13	7.167	13	**7.668**	**8**	**7.665**	**8**	**7.733**	**8**	**7.497**	**12**	**7.545**	**13**
13	sum_ent	5.515	31	5.425	33	**6.320**	31	**5.684**	**28**	5.572	29	5.603	31	5.529	29	5.466	32	**5.606**	**29**	5.441	33	5.512	32	5.470	33	5.442	25	5.440	25	5.370	25	5.541	31	5.435	32
14	ent	5.316	33	5.487	31	**5.988**	33	5.485	32	5.525	32	**5.541**	33	5.474	31	5.397	33	**5.531**	33	5.509	29	5.512	33	5.487	32	5.353	27	5.352	27	5.307	27	5.386	33	5.322	33
15	diff_var	6.743	16	6.398	17	**10.717**	**7**	6.293	19	6.482	16	6.815	16	6.440	16	**7.022**	**16**	6.668	16	6.698	16	6.618	16	6.633	16	6.312	13	6.311	13	6.214	15	6.887	16	6.719	16
16	diff_ent	5.980	21	5.883	24	**8.857**	**14**	6.037	20	6.123	18	5.987	22	6.066	18	6.130	21	**6.132**	**20**	**6.176**	**20**	6.025	21	5.981	22	5.624	21	5.638	21	5.530	22	6.015	21	5.733	21
17	inf_meas1	8.512	10	7.813	12	**11.246**	**4**	8.377	12	7.690	12	7.641	12	7.608	12	8.093	11	7.487	12	7.545	12	7.305	12	7.310	12	**9.524**	**7**	**9.509**	**7**	**9.184**	**7**	8.414	11	7.764	12
18	inf_measure2	12.739	5	14.712	2	10.879	5	15.098	1	**17.137**	**1**	15.148	1	16.202	1	11.473	6	16.503	1	**17.418**	**1**	**16.608**	**1**	**16.608**	**1**	12.370	4	12.366	4	12.509	4	12.951	4	13.920	2
Gray level Difference Statistics (GLGS)																																			
19	hom	5.804	24	5.938	22	**7.150**	**24**	**6.627**	**17**	5.857	25	5.813	25	5.747	25	5.638	26	5.790	26	5.837	25	5.938	23	5.974	24	6.262	15	6.263	15	**6.271**	**14**	5.843	25	5.563	27
20	cont	6.136	20	6.197	19	**9.862**	**11**	5.939	23	6.074	21	6.176	19	6.002	21	6.219	20	6.135	19	**6.222**	**19**	6.174	19	6.135	19	5.733	19	5.732	19	5.636	19	6.161	20	**6.246**	**19**
21	eng	5.828	22	5.818	25	**7.618**	**19**	**6.333**	**18**	6.064	22	6.006	21	5.902	22	5.951	23	**6.088**	**21**	6.069	21	6.007	22	6.047	21	5.668	20	5.661	20	5.605	20	5.931	22	5.620	23
22	ent	5.770	25	5.883	23	**8.618**	**17**	5.865	25	5.905	23	5.923	23	5.798	23	5.970	22	**6.010**	**23**	5.930	23	5.918	25	5.894	25	5.503	23	5.502	23	5.493	23	5.880	23	5.632	22
23	mean	5.633	27	5.773	26	**8.659**	**16**	**5.892**	**24**	5.749	26	5.764	26	5.573	27	**5.813**	**24**	**5.818**	**24**	5.649	26	5.731	26	5.740	26	5.431	26	5.431	26	5.357	26	5.699	26	5.590	24
Fourier Power Spectrum (FPS)																																			
24	radial_sum	6.297	18	6.109	20	6.665	26	5.994	21	6.093	19	6.127	20	6.044	19	6.847	17	6.075	22	6.020	22	6.132	20	6.123	20	**11.725**	**5**	**11.726**	**5**	**11.879**	**5**	**6.198**	**18**	6.190	20
25	ang_sum	10.120	7	10.584	7	9.350	12	**11.538**	**5**	10.527	5	10.218	6	10.533	6	10.203	7	10.471	5	10.370	7	10.216	6	10.175	6	**11.207**	**6**	**11.205**	**6**	**11.492**	**6**	10.211	8	10.798	7
Haar Wavelet Features																																			
26	h_mean	13.910	3	**14.778**	**1**	11.456	3	13.852	3	14.396	2	13.898	3	**14.547**	**2**	14.175	2	14.408	3	14.128	2	13.605	3	13.663	3	14.076	2	14.078	2	14.480	2	14.073	2	**14.599**	**1**
27	h_var	5.652	26	5.474	32	6.219	32	5.432	33	5.505	33	5.575	32	5.497	30	5.501	29	5.533	32	5.474	32	5.514	31	5.489	31	**6.636**	**12**	**6.635**	**12**	**6.597**	**12**	5.668	28	5.548	28
Region Based Features																																			
28	area	9.372	8	9.365	9	9.240	13	**9.506**	**9**	9.368	9	9.239	9	9.065	9	**9.757**	**8**	9.179	9	9.198	9	9.328	9	9.236	9	0.505	29	0.505	32	0.530	33	8.853	9	**9.615**	**8**
29	perim	7.722	12	7.940	11	**8.361**	**18**	**8.449**	**11**	**8.047**	**11**	7.993	11	7.748	11	7.986	12	7.863	11	7.770	11	7.983	11	7.961	11	0.505	30	0.505	29	0.530	30	7.425	15	7.765	11
30	MajorAxisLength	8.991	9	9.835	8	9.982	9	10.143	8	9.680	8	**10.203**	**7**	9.916	8	9.525	9	**10.318**	**7**	9.528	8	10.113	7	10.071	7	0.505	28	0.505	33	0.530	32	**10.356**	**7**	9.036	9
31	EquivDiameter	7.878	11	8.684	10	8.774	15	8.716	10	8.633	10	**8.980**	**10**	8.692	10	8.396	10	**9.075**	**10**	8.425	10	**9.068**	**10**	8.844	10	0.505	31	0.505	30	0.530	28	8.736	10	8.317	10
32	Extent	14.172	2	11.437	5	10.718	6	11.271	6	10.176	7	9.685	8	9.996	7	13.824	3	10.175	8	10.743	6	13.150	4	12.974	4	0.505	32	0.505	31	0.530	29	11.701	6	11.624	6
33	convex_area	12.842	4	**14.691**	**3**	13.174	1	14.480	2	14.346	3	**14.718**	**2**	14.434	3	13.350	4	**14.974**	**2**	14.045	3	14.100	2	14.140	2	0.505	33	0.505	28	0.530	31	13.778	3	13.727	4

Table 3.6: Feature weights (W) and ranking order (O-R) for different feature sets (a total of 33 different features) extracted both from simulated speckeled images (with noise variance $\sigma^2 = 0.05$) and filtered images.

				Median		Homogeneity		Geometric		Local Statistics				Anisotropic Diffusion						Wavelet Based filtering												Hybrid filtering			
		Original Image		Median		Homog		Geom		MeanV		Wiener		PMAD		SRAD		DPAD		DWT		DT-CWT		DT-CWT_S		DT-CWT_H		DT-CWT_T		DT-CWT_B		DLWFDW		HybridMedian	
Feature No	Feature name	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O	W	R-O
First Order Statistics Histogram Features																																			
1	mean	5.245	32	5.540	30	**6.520**	**28**	5.432	33	5.610	30	5.635	28	5.523	32	5.493	32	5.609	30	5.464	33	5.532	30	**6.535**	**21**	**6.326**	**21**	**6.394**	**21**	**6.082**	**17**	5.610	28	5.605	31
2	st	**8.762**	**9**	7.314	15	7.416	21	8.362	**8**	7.173	15	7.080	14	7.388	14	7.432	15	7.223	14	7.272	12	7.137	13	5.983	24	6.092	23	6.081	23	6.351	13	6.681	19	7.529	13
3	skew	13.001	5	12.933	4	10.854	5	12.392	18	12.672	4	12.939	4	11.981	4	12.157	5	12.473	4	13.075	4	12.515	4	**14.918**	**2**	**14.732**	**2**	**14.787**	**2**	13.263	3	11.887	5	12.884	4
4	eng	5.525	26	5.643	24	**6.553**	25	**6.165**	26	5.613	27	5.631	29	5.537	30	5.674	26	5.633	28	5.832	23	5.564	28	6.001	23	5.992	24	**6.015**	**24**	5.980	20	5.450	33	5.712	23
5	ent	5.875	19	5.612	27	**6.537**	26	**6.396**	**3**	5.665	26	5.639	26	5.668	26	5.674	25	5.671	26	**5.811**	25	5.606	27	5.568	26	5.555	27	5.587	27	5.629	22	5.513	31	5.694	24
Spatial Gray Level Dependence Matrices (SGLDM)																																			
6	ang_sec_mom	6.067	18	6.555	16	6.756	24	5.714	17	6.608	16	6.646	16	6.471	16	6.477	17	6.493	17	6.215	16	6.519	17	**8.433**	**9**	**8.272**	**9**	**8.298**	**9**	**7.965**	**7**	6.605	23	6.299	17
7	cont	5.635	24	5.998	19	9.252	14	5.820	30	5.863	22	5.999	21	5.789	21	5.715	21	5.997	19	6.137	17	6.057	19	**7.260**	**10**	**7.548**	**10**	**7.629**	**10**	6.116	14	7.245	14	5.850	19
8	corr	8.434	12	10.392	7	11.919	2	9.514	32	9.857	6	8.815	10	10.874	5	9.692	7	9.583	8	7.073	14	7.879	10	**15.363**	**1**	**15.794**	**1**	**15.591**	**1**	**17.554**	**1**	**16.159**	**1**	9.074	7
9	sum_sq	**8.730**	10	7.329	14	7.427	20	8.333	25	7.187	14	7.079	15	7.402	13	7.451	13	7.236	13	7.272	13	7.116	14	6.187	22	6.277	22	6.250	22	6.494	12	6.681	20	7.600	**12**
10	inv_diff_mom	5.642	23	5.993	21	**6.484**	30	5.962	28	5.957	19	6.038	19	5.859	19	6.019	20	5.906	21	5.534	30	5.975	21	**6.587**	**18**	**6.447**	**18**	**6.452**	**18**	6.040	19	6.390	24	5.800	22
11	sum_avg	5.246	31	5.541	29	**6.521**	27	5.421	15	5.611	29	5.635	27	5.524	31	5.493	31	5.609	29	5.465	32	5.532	29	**6.544**	**20**	**6.334**	**20**	**6.403**	**20**	6.091	**16**	5.610	27	5.606	30
12	sum_var	8.912	7	7.378	13	7.474	19	8.479	31	7.232	13	7.125	13	7.480	12	7.618	12	7.280	12	7.340	11	7.203	12	6.817	14	6.898	12	6.872	13	7.088	10	6.771	17	7.657	11
13	sum_ent	5.790	20	5.453	32	**6.493**	29	6.387	29	5.591	31	5.562	31	5.551	29	5.577	29	5.595	31	5.753	27	5.469	33	5.602	25	5.580	26	5.612	25	5.646	21	5.465	32	5.597	32
14	ent	5.197	33	5.421	33	**6.222**	33	5.488	**7**	5.541	33	5.520	33	5.428	33	5.486	33	5.530	33	5.612	29	5.524	31	5.539	27	5.624	25	5.590	26	5.598	23	5.537	30	5.552	33
15	diff_var	6.368	17	6.445	17	**9.535**	**11**	6.422	20	6.451	17	6.576	17	6.423	17	6.161	18	6.692	16	6.531	15	6.641	16	**8.767**	**8**	**9.033**	**8**	**8.993**	**8**	7.817	8	8.297	11	6.326	16
16	diff_ent	5.450	27	5.629	26	**8.874**	**16**	5.604	16	5.753	24	5.772	24	5.752	23	5.674	27	5.804	24	5.878	21	5.844	24	**6.857**	**13**	6.543	15	**6.587**	**15**	5.557	26	**6.777**	**16**	5.627	28
17	inf_meas1	6.869	15	7.729	12	**10.298**	**6**	8.072	22	7.272	12	7.587	12	7.022	15	7.443	14	7.060	15	6.075	19	6.758	15	**12.133**	**3**	**11.003**	**6**	**11.495**	**6**	7.394	9	**10.311**	**7**	6.837	15
18	inf_measure2	14.506	3	17.297	1	11.336	3	14.936	23	**18.223**	**1**	**17.510**	**1**	**17.891**	**1**	**18.038**	**1**	**17.624**	**1**	**19.466**	**1**	**18.017**	**1**	10.932	6	12.289	3	11.981	3	14.898	2	10.698	6	**18.431**	**1**
Gray level Difference Statistics (GLGS)																																			
19	hom	5.642	22	5.993	22	6.484	31	5.962	12	5.957	20	6.038	20	5.859	20	6.019	19	5.906	22	5.534	31	5.975	22	**6.587**	**19**	**6.447**	**19**	**6.452**	**19**	6.040	18	6.390	25	5.800	21
20	cont	5.635	25	5.998	20	**9.252**	**15**	5.820	9	5.863	23	5.999	22	5.789	22	5.715	22	5.997	20	6.137	18	6.057	20	**7.260**	**11**	**7.548**	**11**	**7.629**	**11**	6.116	15	7.245	15	5.850	20
21	eng	5.281	30	5.740	23	**7.056**	22	5.632	2	5.884	21	5.845	23	5.738	24	5.694	23	5.858	23	5.814	24	5.915	23	**6.772**	**16**	**6.837**	**13**	**6.808**	**14**	5.577	24	6.659	21	5.666	26
22	ent	5.443	28	5.629	25	**8.424**	**17**	5.474	21	5.731	25	5.758	25	5.738	25	5.689	24	5.784	25	5.854	22	5.787	25	**6.777**	**15**	**6.517**	**16**	**6.538**	**16**	5.568	25	**6.693**	**18**	5.618	29
23	mean	5.376	29	5.605	28	**8.404**	**18**	5.484	24	5.642	26	5.619	30	5.627	27	5.595	28	5.646	27	5.789	26	5.673	26	**6.597**	**17**	**6.504**	**17**	**6.458**	**17**	5.513	27	**6.650**	22	5.644	27
Fourier Power Spectrum (FPS)																																			
24	radial_sum	6.752	16	6.109	18	6.926	23	6.954	**6**	6.135	18	6.096	18	6.097	18	7.134	16	6.098	18	6.047	20	6.098	18	**11.967**	**4**	**11.719**	**4**	**11.734**	**4**	**11.633**	**5**	6.101	26	6.221	18
25	ang_sum	11.352	6	10.453	6	9.695	9	**12.269**	**4**	10.526	5	10.305	6	10.644	6	10.642	6	10.436	6	10.311	6	10.080	7	9.934	7	9.999	7	9.948	7	10.614	6	9.571	8	10.723	6
Haar Wavelet Features																																			
26	h_mean	15.381	1	14.431	2	11.310	4	**15.406**	**1**	14.091	3	14.374	3	14.432	2	14.159	2	14.104	3	13.999	2	13.284	3	11.953	5	11.576	5	11.577	5	12.259	4	12.555	4	14.662	2
27	h_var	5.751	21	5.474	31	6.420	32	5.766	5	5.570	32	5.551	32	5.575	28	5.560	30	5.570	32	5.715	28	5.512	32	**6.947**	**12**	**6.824**	**14**	**6.874**	**12**	**6.713**	**11**	5.561	29	5.678	25
Region Based Features																																			
28	area	8.776	8	**9.243**	**9**	9.548	10	8.661	11	**9.255**	**9**	**9.198**	**8**	8.905	9	**9.129**	**8**	9.064	9	8.939	8	9.093	9	0.449	33	0.448	29	0.444	33	0.454	32	8.902	10	8.871	9
29	perim	7.242	14	7.815	11	**9.351**	**13**	7.892	13	**7.963**	**11**	**7.945**	**11**	7.699	11	7.676	11	7.772	11	7.564	10	7.804	11	0.449	28	0.448	33	0.444	28	0.454	28	**7.890**	**13**	7.356	14
30	MajorAxisLength	8.655	11	9.428	8	10.268	7	9.868	10	9.548	8	**10.383**	**5**	9.886	7	8.848	9	10.**482**	5	9.091	7	**10.744**	**6**	0.449	31	0.448	28	0.444	29	0.454	29	9.016	9	8.951	8
31	EquivDiameter	7.630	13	8.320	10	**9.506**	**12**	8.043	19	8.481	10	**9.036**	**9**	8.507	10	8.005	10	9.020	10	8.184	9	**9.399**	**8**	0.449	32	0.448	30	0.444	30	0.454	33	8.295	12	8.123	10
32	Extent	14.781	2	10.719	5	10.025	8	10.524	27	9.812	7	9.223	7	9.865	8	12.735	4	9.607	7	10.750	5	11.883	5	0.449	29	0.448	32	0.444	31	0.454	30	14.462	2	10.873	5
33	convex_area	13.242	4	14.243	3	13.704	1	14.367	14	14.385	2	**15.003**	**2**	**14.425**	**3**	13.237	3	**14.989**	**2**	13.896	3	**14.509**	**2**	0.449	30	0.448	31	0.444	32	0.454	31	12.940	3	13.706	3

Table 3.7: Improved feature ranking order (R-O) scores for 5 feature sets (FOS, SGLDM, GLGS, FPS and Haar wavelets) applied on original 286 PF US images using different despeckling methods.

		Median	Homogeneity	Geometric	Local Statistics		Anisotropic Diffusion			Wavelet Based filtering						Hybrid filtering	
Feature set	No. of features	Median	Homog	Geom	MeanV	Wiener	PMAD	SRAD	DPAD	DWT	DT-CWT	DT-CWT_S	DT-CWT_H	DT-CWT_T	DT-CWT_B	DLWFDW	HybridMedian
FOS	5	3	2	3	3	3	3	3	3	3	3	3	5	5	5	3	3
SGLDM	13	6	5	5	4	4	5	0	4	6	5	5	12	12	12	4	3
GLGS	5	4	4	2	3	4	1	2	4	4	3	3	5	5	5	2	3
FPS	2	0	0	1	1	1	1	1	1	0	1	1	2	2	2	0	0
Haar wavelets	2	0	0	0	1	0	1	1	0	1	0	0	2	2	2	1	1
Total Score		13	11	11	12	12	11	7	12	14	12	12	26	26	26	10	10

Table 3.8: Improved feature ranking order (R-O) scores for 5 feature sets (FOS, SGLDM, GLGS, FPS and Haar wavelets) applied on 286 noise simulated PF US images using different despeckling methods.

		Median	Homogeneity	Geometric	Local Statistics		Anisotropic Diffusion			Wavelet Based filtering						Hybrid filtering	
Feature set	No. of features	Median	Homog	Geom	MeanV	Wiener	PMAD	SRAD	DPAD	DWT	DT-CWT	DT-CWT_S	DT-CWT_H	DT-CWT_T	DT-CWT_B	DLWFDW	HybridMedian
FOS	5	3	2	2	2	2	1	0	2	2	2	3	3	3	3	1	4
SGLDM	13	8	6	3	7	8	6	6	8	6	10	9	9	9	10	6	7
GLGS	5	4	4	5	5	4	5	5	4	4	4	5	5	5	5	5	5
FPS	2	0	0	2	1	0	0	0	0	0	0	1	1	1	1	0	0
Haar wavelets	2	0	0	2	0	0	0	0	0	0	0	1	1	1	1	0	0
Total Score		15	12	14	15	14	12	11	14	12	16	19	19	19	20	12	16

3.6.4 Visual evaluation by medical experts

Tables 3.9, 3.10 and Figures 3.6, 3.7 summarize the results of the visual evaluation of the original and the filtered images carried out by two medical experts. Tables 3.9 and 3.10 show the results of the total visual scoring, the percentage scoring assigned by both experts, and the inter-operator variability between the two experts using linear regression and paired t-test statistical analysis for original unmodified images and simulated speckled images, respectively. Figures 3.6 and 3.7 represent the average image quality scoring by Expert 1 and Expert 2, and the overall scoring percentages for different 16 despeckling methods applied on original unmodified images and simulated speckled images, respectively.

For Table 3.9 and Figure 3.6, the average image quality scores obtained by Expert 1 revealed that the best despeckling filter is DT-CWT_S with a score of 67% followed by and DT-CWT_H, DT-CWT_T, DLWFDW, DT-CWT_B, HybridMedian, SRAD and MeanV with scores of 60%, 60%, 60%, 60%, 53% and 53%, respectively. On the other hand, Expert 2 scoring suggested that the best despeckling filter is DT-CWT_H with a high score of 80% followed by DT-CWT_S, DT-CWT_T, DT-CWT_B, DLWFDW and HybridMedian with scores of 60%, 60%, 60%, 60% and 47%, respectively. The overall average percent scoring by both experts revealed that: (1) the highest average percent score was assigned to the filters DT-CWT_H and DT-CWT_S with scores of 70% and 63%, respectively, followed by DT-CWT_T, DT-CWT_B, DLWFDW, and HybridMedian with scores of 60%, 60%, 60% and 50%, respectively; and (2) the lowest overall percent score was assigned to the filters Geom, Homog, Median and DT-CWT_B with scores of 20%, 27%, 27%, 30%, and 37%, respectively.

Regarding the second results using the second dataset (simulated speckle noise images) as shown in Table 3.10 and Figure 3.7, the high average scores obtained by Expert 1 were assigned to filters DT-CWT_S and DT-CWT_H with a score of 80% followed by DT-CWT_T, DPAD, DLWFDW, SRAD, Wiener, DT-CWT_B, DT-CWT, DWT, PMAD, MeanV and HybridMedian with scores of 73%, 73%, 73%, 67%, 67%, 67%, 60%, 60%, 60%, 60% and 53% respectively. For Expert 2, the high score was given to filters DT-CWT_S and DT-CWT_H with a score of 67% followed by DT-CWT_T, DLWFDW, DT-CWT, HybridMedian, DWT, PMAD, MeanV, Wiener and Homog with scores of 60%, 60%, 60%, 60%, 53%, 53%, 53%, 53% and 40%, respectively. For the overall average percent scores, (1) the best scores were assigned to filters DT-CWT_S and DT-CWT_H with a score of 73% followed by DT-CWT_T, DPAD, DLWFDW, Wiener, DT-CWT, SRAD, HybridMedian, PMAD and MeanV with scores of 67%, 63%, 63%, 63%, 60%, 60%, 57%, 57%, 57%, 57% and 57%, respectively; and (2) the lowest overall percent scores were assigned to filters Geom, Homog, Median and DT-CWT_B with scores of 33%, 37%, 40% and 47%, respectively.

The inter-operator variability test using linear regression and paired *t*-test statistical analysis presented in Tables 3.9, 3.10 and Figure 3.8 revealed a significant positive pairing between Expert 1 and Expert 2 for all average image quality scoring using different despeckling methods. For the first regression analysis presented in 3.9 concerning the first unmodified PF US dataset images, $R^2 = 0.68$ with $p < 0.0001$ was obtained, showing fair to good agreement between the two experts. This is indicating a slight disagreement in the visual image quality scoring between the two experts. This is because of the absence of the ground truth or a noise-free reference images, as it is a little bit hard for both experts to assess visually the filtered images using only the original image (assumed noisy) without a ground truth or noise-free image. The same things for the quantitative evaluation metrics (using the first dataset) described earlier in Section 3.6.2 are basically defined between the original image (assumed noisy) and de-speckled ones.

Table 3.9: The average visual scoring using three perception criteria and inter-operator variability between the two medical experts (using statistical analysis) for all the despeckling methods applied directly to the first set of unmodified PF US images.

Experts' Scoring / Filters	Median	Homog	Geom	MeanV	Wiener	PMAD	SRAD	DPAD	DWT	DT-CWT	DT-CWT_S	DT-CWT_H	DT-CWT_T	DT-CWT_B	DLWFDW	HybridMedian
Expert 1																
Global Image Quality	1	1	1	3	2	2	3	2	2	2	3	3	3	3	3	3
PF Difinition	1	2	2	3	2	2	3	2	2	2	3	3	3	3	3	3
Edge Preservation	1	2	2	2	2	1	2	1	2	2	4	3	3	3	3	2
Total scoring /15	3	5	5	8	6	5	8	5	6	6	10	9	9	9	9	8
Average (%)	20	33	33	53	40	33	53	33	40	40	67	60	60	60	60	53
Expert 2																
Global Image Quality	1	1	1	2	1	2	2	2	2	2	3	4	3	3	3	3
PF Difinition	1	1	1	2	1	2	2	2	2	2	3	4	3	3	3	2
Edge Preservation	1	1	1	2	1	2	2	2	2	2	3	4	3	3	3	2
Total scoring /15	3	3	3	6	3	6	6	6	6	6	9	12	9	9	9	7
Average (%)	20	20	20	40	20	40	40	40	40	40	60	80	60	60	60	47
Overall Average (%)	20	27	27	47	30	37	47	37	40	40	63	70	60	60	60	50
Inter-operator variability (Expert 1 vs Expert 2)																
Linear regression analysis:	**R**	0.826		**R**^{2}	0.682		**P**	0.00008		**P** < 0.0001						
Paired *t*-test	**P**	0.487		**P** > 0.025												

Table 3.10: The average visual scoring (using three perception criteria) and statistical regression analysis between the two medical experts for all despeckling filters used in this study

Experts' Scoring / Filters	Median	Homog	Geom	MeanV	Wiener	PMAD	SRAD	DPAD	DWT	DT-CWT	DT-CWT_S	DT-CWT_H	DT-CWT_T	DT-CWT_B	DLWFDW	HybridMedian
Expert 1																
Global Image Quality	2	2	1	3	3	3	3	3	3	3	4	4	3	3	4	3
PF Difinition	2	2	2	3	4	3	3	4	3	3	4	4	4	3	3	3
Edge Preservation	2	1	2	3	3	3	4	4	3	3	4	4	4	3	3	2
Total scoring / 15	6	5	5	9	10	9	10	11	9	9	12	12	11	9	10	8
Average (%)	40	33	33	60	67	60	67	73	60	60	80	80	73	60	67	53
Expert 2																
Global Image Quality	2	2	1	2	3	3	2	3	3	3	3	4	3	1	3	3
PF Difinition	2	2	2	3	3	3	3	2	2	3	4	3	3	2	3	3
Edge Preservation	2	2	2	3	3	2	3	3	3	3	3	3	3	2	3	3
Total scoring /15	6	6	5	8	9	8	8	8	8	9	10	10	9	5	9	9
Average (%)	40	40	33	53	60	53	53	53	53	60	67	67	60	33	60	60
Overall Average (%)	40	37	33	57	63	57	60	63	57	60	73	73	67	47	63	57
Inter-operator variability (Expert 1 vs Expert 2)																
Linear regression analysis:	**R**	0.925		**R**^{2}	0.855		**P**	0.0000003		**P** < 0.0001						
Paired *t*-test:	**P**	0.228		**P** > 0.025												

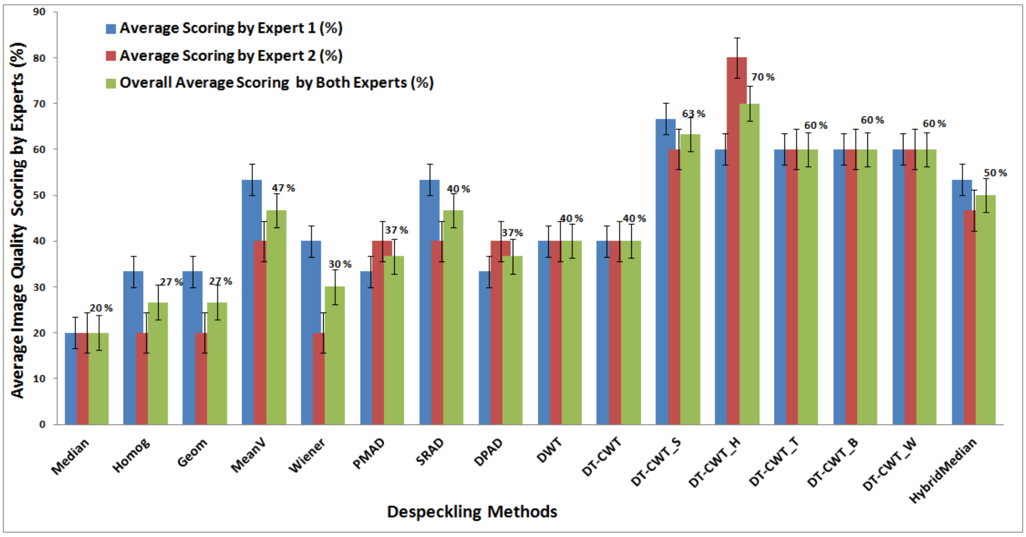


Figure 3.6: Average image quality scoring by Expert 1 and Expert 2, and the overall average scoring percentages for different despeckling filters applied directly on the first set of unmodified PF US images.

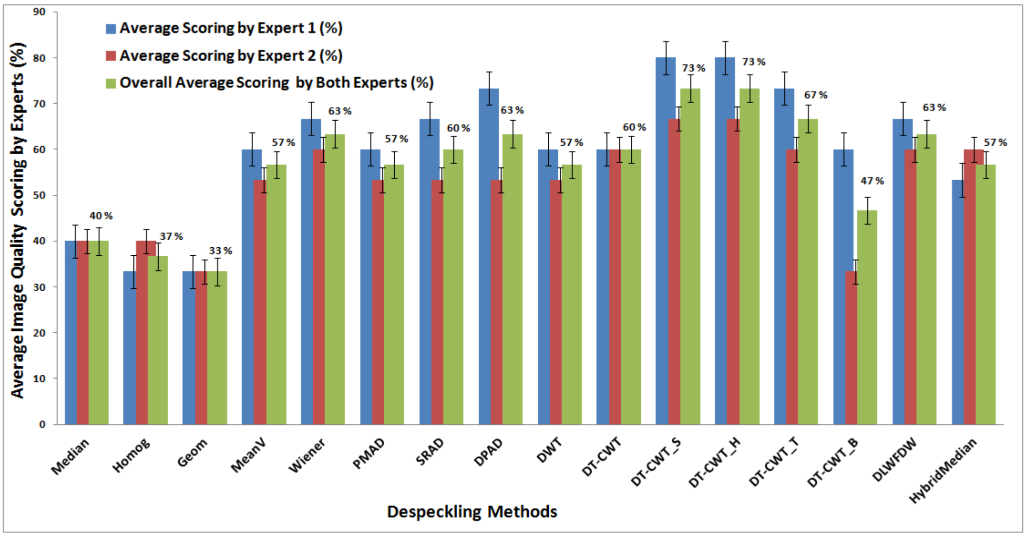
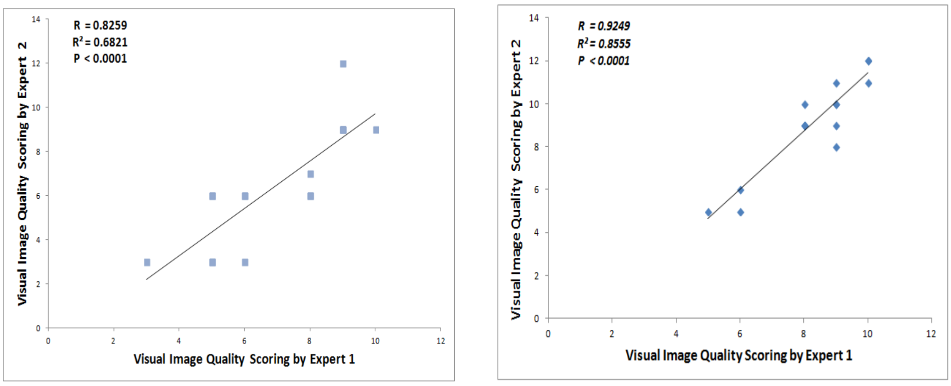


Figure 3.7: Average image quality scoring by the two experts and the overall scoring percentages for different despeckling methods applied on the second dataset using simulated speckle noise.



(a) **(b)**

Figure 3.8: Inter-operator variability test using linear regression for different visual image quality scoring performed by Expert 1 and Expert 2 on: (a) the first set of unmodified PF US images ($R^2 = 0.68$) and (b) Simulated speckle noise PF US images, indicating consistent pairing and very good agreement ($R^2 = 0.86$). For the first plot (a), $R2 = 0:68$ with $p < 0:0001$ showing fair agreement between the two experts. This indicates a slight disagreement in the visual image quality scoring between the two experts. This is because of the absence of the ground truth or a noise-free reference images, as it is a little bit hard for both experts to assess visually the filtered images using only the original (assumed noisy) image without a ground truth or noise-free image.

Some researchers may argue that this treatment is not acceptable, imagining that doing nothing on the original image would give the good scores in many of the metrics (such as MSE, PSNR, AD, etc). So, preparing a set of ground-truth data, adding artificial speckle noise, and comparing the de-speckled images with the ground-truths would be more appropriate for the evaluation study. For this reason, we introduced the second simulated speckle noise dataset in this study, to highlight the effect of the simulated multiplicative speckle noise on the global image quality improvement and edge preservation. For the second regression analysis presented in 3.10 concerning the speckle noise simulated PF US images, $R^2 = 0.86$ with $p < 0.0001$ was obtained, indicating excellent agreement between the two experts. By checking the visual results for Figures 3.2,3.3, 3.4 and 3.5, the overall image quality scoring of Tables 3.9, 3.10 and Figures 3.6, 3.7 and the statistical analysis shown in Tables 3.9, 3.10 and Figure 3.8, we can conclude that filters DT-CWT_S, DT-CWT_H, DT-CWT_T, DPAD, DLWFDW, Wiener, DT-CWT, SRAD, HybridMedian, PMAD and MeanV are accepted for clinical practice. However, other filters with lowest overall percent score such as Geom, Homog, Median unacceptable for clinical use.

3.7 Conclusion

Analysing medical US images is difficult because of the presence of speckle noise, which reduces image contrast, destroys or diffuses the image edges and affects the delineation of PF. It also affects the prediction and the detection of low contrast objects with small lesions, making medical interpretation and biometric measurements difficult, and therefore impacting the efficiency of diagnosis. Furthermore, this effect may alter the performance of some medical post-processing applications such as edge detection, registration, feature extraction, feature selection, automated segmentation and pattern recognition. Therefore, in order to facilitate these medical image processing application and make them more effective, pre-processing techniques should be applied to the speckled datasets. These techniques are very important in the filtering of different PF US images in terms of improving the global image visibility, image edges and its usefulness after the effect of the acoustic speckle noise. In this study, a comparative evaluation attempt has been made to analyse 7 groups and 16 existing speckle reduction methods. This includes: Median, adaptive local statistic filters (MeanV and Wiener); homogeneity (Homg); geometric (Geom); anisotropic diffusion (PMAD, SRAD and DPAD); wavelet-based filtering using universal and soft threshoding rules (DWT and DT-CWT), and DT-CWT filters using BayesShrink thresholding method and different thresholding fuctions (hard, soft, trimmed, bivariate and wiener); and Hybrid filters (DLWFDW and HybridMedian). The evaluation protocol approach was based on the quantitative image quality metrics, feature ranking and selection analysis, and visual evaluation by two medical experts. The experimental result of this study has shown that the filters based on dual tree complex wavelet transform (DT-CWT) using BayesShrink subband thresholding and different thresholding functions namely, hard, soft, trimmed and bivariate (DT-CWT_S, DT-CWT_H, DT-CWT_T and DT-CWT_B) can be introduced successfully for the filtering and the processing of PF US images. These filters present a superior edge preserving behaviour, and their filtering results have shown good visual appearance in our experiments. It is also envisaged from this study that these filters can be used as a preprocessing step for the automated segmentation of the PF region, followed by PF texture analysis, and classification. However, further investigation is required to (i) evaluate the performance of these filters on a larger dataset of US images (normal and abnormal PF US images) as well as their usefulness in the medical practice, and (ii) analyse its impact on medical applications by using different US machines (e.g. portable and mobile US imaging systems) with advanced specifications.

Chapter 4

Plantar Fascia Segmentation and Thickness Estimation in Ultrasound Images [*]

4.1 Overview

This chapter considers an automatic segmentation proposed approach which for the first time extracts ultrasound data to estimate size across three sections of the PF (rearfoot, midfoot and forefoot). This segmentation method uses artificial neural network module (ANN) in order to classify small overlapping patches as belonging or not-belonging to the region of interest (ROI) of the PF tissue. Features ranking and selection techniques were performed as a post-processing step for features extraction to reduce the dimension and number of the extracted features. The trained ANN classifies the image overlapping patches into PF and non-PF tissue, and then it is used to segment the desired PF region. The PF thickness was also calculated using two different methods: distance transformation and area-length calculation algorithms. The proposed approach and other segmentation comparison methods' results are also discussed.

[*] This Chapter is based on a published journal paper in Computerized Medical Imaging and Graphics (Boussouar et al., 2017a)

4.2 Introduction

Automatic segmentation is one of the most critical tasks in medical image analysis; it is mainly used to locate region of interest (ROI) objects and boundaries in images. It is considered the most challenging task in medical US imaging compared to other imaging modalities, such as CT and MRI due to attenuation, speckles, shadows, signal loss and drop-out. Furthermore, there is no commonly accepted method for US image segmentation because segmentation techniques vary widely according to the specific problem, application, imaging modality, human interaction, the homogeneity of images, spatial characteristics of images, continuity, texture and image content (Noble and Boukerroui, 2006; Rueda et al., 2014). Although many segmentation methods and techniques of US images exist, there is little literature on the segmentation process of the plantar fascia in US images of the foot. The only previous work found in relation to PF tissue US images is that reported in (Deshpande et al., 2013) using the Chan-Vese active contour segmentation method (Chan and Vese, 2001). The Chan-Vese model is based on the variational information in grayscale intensities of the image. This proposed technique was effective in the detection of bones and in segmenting the soft tissue layers between the bone and the skin in US images of the foot. However, this method is used for segmenting the whole plantar tissue without defining different plantar tissue areas. Most active contour methods used in US images suffer from the following shortcomings that seriously affect the segmentation results (Chang et al., 2010): (1) these methods are sensitive to the edge gradient; (2) they need a clear definition of the initial contour mask; (3) they depend on the number of iterations which may affect segmentation accuracy; and (4) they suffer from a high level of computational complexity. Many researchers have made various improvements to the standard active contour, but the disadvantages of this method are still not fundamentally overcome.

Artificial neural network (ANN) techniques have attracted considerable attention in medical imaging due to its intelligence and learning capabilities of performing complicated tasks such as US segmentation and classification. Previous studies (Chang et al., 2010; Noble and Boukerroui, 2006) have shown that integration of ANN can facilitate and improve the segmentation process. Figure 4.1 illustrates how ANNs can be used to segment the ROI of US images. In general, ANNs supervised segmentation approaches consist of following steps: (1) the input images are divided into different overlapping patches (i.e. PF images were divided into small and square distinct blocks of size 9×9 that overlap each other using an overlap of 4.5 pixels); (2) different sets of features are calculated on these image patches and then selected to reduce their redundancy; (3) the selected feature vectors are then presented as input vectors to the trained ANN (trained previously with a set of ground truth segmentation, performed manually by experts) where the image patches are classified as a part of either the background or the ROI; (4) the results of the image patches classification are then combined and merged into a region mask (in black and white colour for background and ROI, respectively); (5) region mask labelling and superposing.

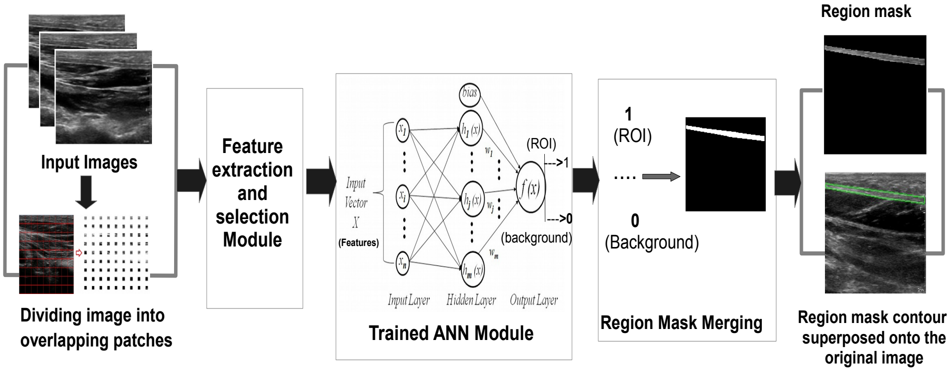


Figure 4.1: Block diagram showing ANNs approach to segmenting ROIs

The manual segmentation and analysis of the large PF US datasets is a tedious, time-consuming and complex task for physicians and clinicians, who have to manually select the ROIs and extract useful diagnostic information. This analysis will lead to inter- or intra-operator variability errors. Motivated by the advantages offered by ANN approaches, we propose a general segmentation ANN-based approach that uses the Radial Basic Function Neural Network (RBF-NN) classifier (Ham and Kostanic, 2000) to automatically segment, estimate PF thickness, to improve PF US data analysis and to assist doctors in qualitative diagnosis. Six different textual feature sets extracted from the ROI are used to train the RBF-NN. The trained RBF-NN classifies PF patches into PF ROIs and background (non-PF), and then is used to segment the PF region. PF thickness is calculated using two different approaches: distance transformation and area-length calculation. This is, to our knowledge, the first segmentation method in the plantar fascia

US imaging field. Therefore, accuracy of the technique at this stage is an important step to facilitate the success of the classification process during clinical diagnosis.

4.3 Proposed plantar fascia segmentation and thickness estimation model

The proposed model consists of the following steps as illustrated in Figure 4.2: (1) pre-processing: during this stage, speckle noise reduction and enhancement filters are applied, then images are divided into small overlapping patches; (2) feature extraction, ranking and selection (feature analysis): in this stage, 32 different features are extracted from the ROIs training dataset and analysed so that they are more prominent and suitable for RBF-NN classifier using feature ranking and selection techniques; (3) training and testing RBF-NN: the RBF neural network classifier is trained using the selected training dataset and classifies the overlapping patches to PF and non-PF regions; (4) PF segmentation and thickness estimation: segmentation is carried out using the trained RBF-NN and PF thickness is calculated using two methods; (a) distance transformation with median calculation; and (b) average thickness expressed as PF area divided by PF length.

4.3.1 PF US image acquisition and tools

Different plantar fascia US images, scanned from a patient's footprint area (in the prone position) were used in this study (Figure 4.3); 150 different real US images were collected from 25 patients to compare the presented methods (6 PF US images per patient for different PF structures rearfoot, midfoot and forefoot sections) with 256 gray levels, a size dimension of 600×655 pixels and a resolution of 28.35 pixels/centimeter. All the proposed method stages were implemented using Matlab R2016a (The MathWorks Inc., Natwick, USA).

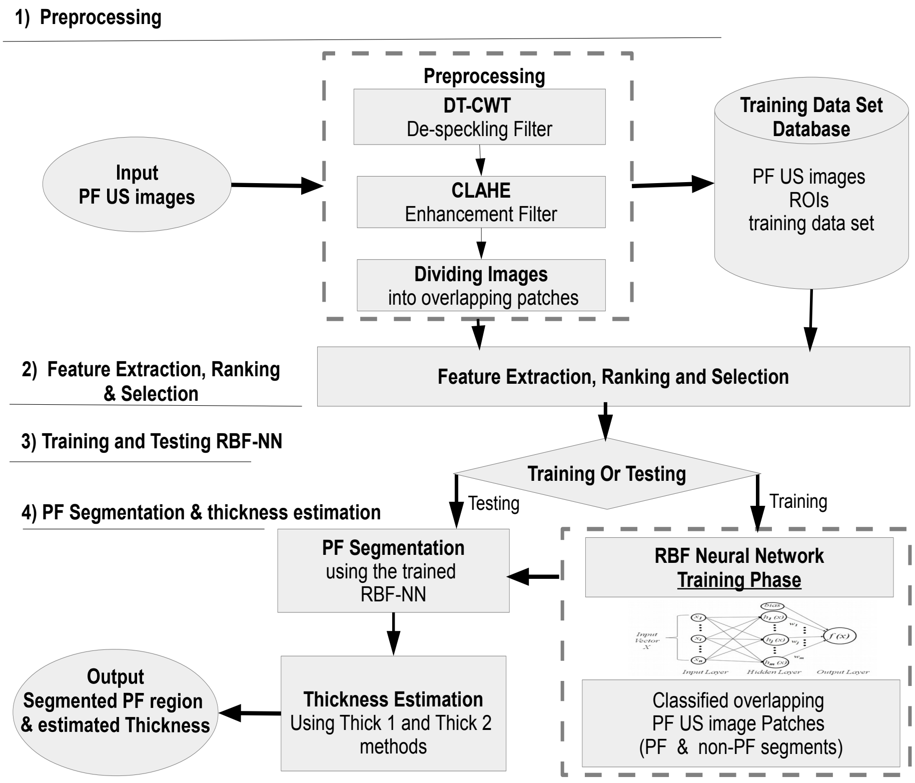


Figure 4.2: Plantar Fascia segmentation and thickness estimation in ultrasound images approach

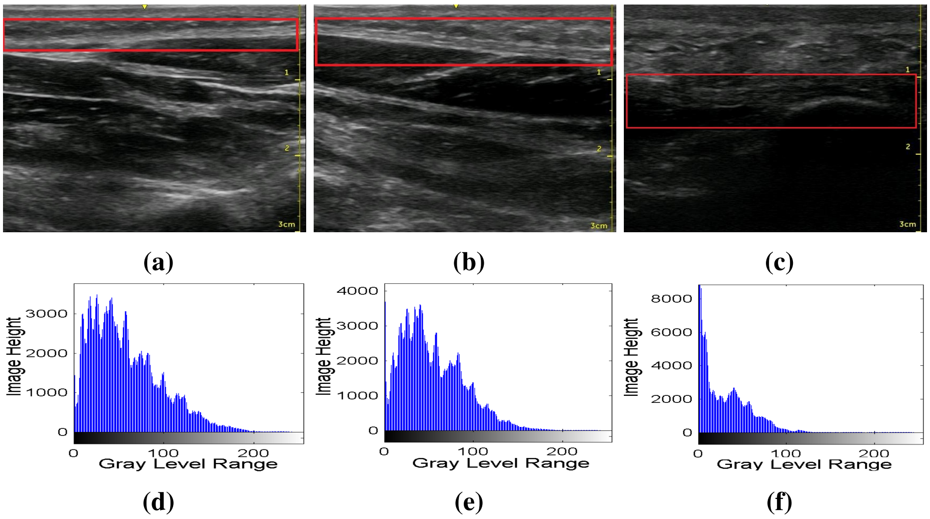
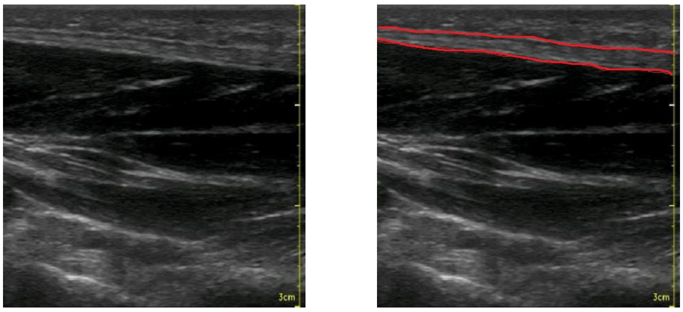


Figure 4.3: (a)-(c) US images for different PF structures: (a) Forefoot, (b) Midfoot and (c) Rearfoot section. (d)-(f) Gray level histogram representation

These images were obtained from the Health Sciences Department, University of Salford, acquired by two expert clinicians according to a precise protocol using a portable Venue 40 musculoskeletal US system (GE Healthcare, UK) with a $5-13$ MHz wideband linear array probe $(12.7\ mm \times 47.1\ mm)$. The thickness of the PF was measured manually (Figure 4.4) at three different sites: rearfoot, midfoot and forefoot sections .



(a) (b)

Figure 4.4: (a) Original image and (b) Targeted PF region selected by a physician (red contours)

4.3.2 Preprocessing

The presence of speckle noise in medical US images is a very common, undesirable feature as it significantly degrades image quality, thereby decreasing the efficiency and reliability of medical image processing tasks, such segmentation and feature extraction. Thus, despeckling and enhancement preprocessing steps are employed to reduce noise and improve the visual quality of the acquired PF images, followed by dividing the images into overlapping patches. Preprocessing steps are described in more detail in the following sub-sections.

4.3.2.1 Dual-tree complex wavelet transform (DT-CWT) filter

Motivated by the advantages of DT-CWT (Kingsbury, 1998), a dual tree complex wavelet transform filter was applied before the CLAHE algorithm to reduce speckle noise, enhance PF images and avoid noise amplification in US images. This filter integrates homomorphic transformation and multi-scale DT-CWT to reduce speckle noise in US images. Implementation details of the DT-CWT can be found in (Kingsbury, 1998, 1999; Selesnick et al., 2005). The despeckling filter consists of the following steps: a) homomorphic transformation; b) DT-CWT image decomposition; c) threshold estimation, modification and suppression of noisy coefficients using BayesShrink thresholding rule (Chang et al., 2000) and bivariate function (Sendur and Selesnick, 2002a); d) application of inverse DT-CWT for signal composition; and e) exponential transformation to obtain despeckled signal.

4.3.2.2 Image enhancement using contrast-limited adaptive histogram equalization (CLAHE)

In medical imaging CLAHE (Zuiderveld, 1994) has proven to be successful for enhancement of low-contrast images. CLAHE is based on the adaptive histogram equalization (AHE) (Pizer et al., 1987) where the histogram is calculated for the contextual region of a pixel.

CLAHE overcomes the limitations of standard histogram equalization and AHE, by calculating the global histogram of an entire image and limiting the contrast. The CLAHE splits the original image into contextual regions, where histogram equalization was applied on each one. The neighbouring sub regions (tiles) are combined by using a bi-linear interpolation to avoid artifact. This could improve the contrast and gives efficient results (Zhao et al., 2010; Lu et al., 2010).

The histogram of a digital image can be defined by the following discrete function with intensity levels in the range $[0, L-1]$,

$$H(r_k) = n_k, \tag{4.1}$$

where r_k is the $k^t h$ intensity value and n_k is the number of pixel in the image with intensity r_k. The normalized histogram is calculated using the following probability density function (PDF),

$$P_r(r_k) = \frac{n_k}{MN}, \qquad k = 0, 1, 2, ..., L-1, \tag{4.2}$$

where $P_r(r_k)$ is an estimated of the probability of occurrence of intensity level r_k in an image. $M \times N$ represents the total number of pixels in the image. The sum of all PDF components is equal to 1. The histogram equalization is obtained by next equation:

$$S_k = (L-1)\sum_{j=0}^{k} P_r\left(r_j\right), \qquad k = 0,1,2,...,L-1, \tag{4.3}$$

where S_k is the new distribution of the histogram.

Owing to the AHE aforementioned issues, in this work the CLAHE enhancement is performed (to adjust the intensity of the PF region) after the speckle noise reduction filter in order to prevent speckle noise amplification and to improve the segmentation results.

The CLAHE consists of the following steps: (1) the original image is split into several non-overlapping regions; (2) the histogram equalization is then applied on each region ; (3) the contrast expansion clip limit is calculated using equation (19) for clipping the histograms; (4) each histogram is redistributed according to the calculated clip limit; (5) gray-scale mapping of the resultant limited histograms, using cumulative distribution function (CDF). These steps are more explained in (Zuiderveld, 1994). The clip limit is calculated using:

$$\beta = \frac{MN}{L}\left(1 + \frac{\alpha}{100}\left(S_{max} - 1\right)\right), \tag{4.4}$$

where β and α are clip limit and clip factor respectively, if $\alpha = 0$, the clip limit equal to $\left(\frac{MN}{L}\right)$, furthermore if $\alpha = 100$, the maximum allowable slope is s_{max}.

4.3.3 Feature extraction

PF area has a reasonably defined structure, with the most common characteristic being its thickness and texture; therefore, features extracted in this work were used to define the shape of the PF region precisely. Six different texture feature sets (a total of 32 features as presented in Table 4.1) were extracted from the overlapping patches. In this work, the Gray level difference statistics (GLDS) were computed for the following displacements: $\delta = (0,1),(1,1),(1,0),(1,-1)$, where $\delta \equiv (\Delta x, \Delta y)$ and their average values were calculated. The Haralick spatial gray level dependence matrices (SGLDM) features were calculated and averaged for a selected distance $d = 1$ (3×3 matrices) and four different orientation angles $\theta = 0^o, 45^o, 90^o$, and 135^o.

Table 4.1: Feature extraction measures

Feature extraction technique	Feature measures	References
1) Histogram features	(1) mean, (2) standard deviation, (3) skewness, (4) energy, and (5) entropy.	(Umbaugh, 2005)
2) Haar wavelet features	(6) mean and (7) variance haar wavelet features of the low-low (LL) frequency sub-band	(Wen et al., 2007a) (Gonzalez and Woods, 2011)
3) BDIP	(8) Block-difference of inverse probabilities measure, to assess variations in local brightness.	(Chun et al., 2003)
4) Gray level difference statistics (GLDS)	(9) contrast, (10) angular second moment, (11) entropy, and (12) mean.	(Weszka et al., 1976a)
5) Haralick spatial gray level dependence matrices (SGLDM)	(13) angular second moment, (14) contrast, (15) correlation, (16) sum of squares, (17) variance, (18) inverse difference moment (InvDiffMoment), (19) sum average, (20) sum variance, (21) sum entropy, (22) entropy, (23) difference variance, (24) difference entropy, and (25) information measures of correlation.	(Haralick et al., 1973)
6) Region based features	(26) area, (27) perimeter, (28) major axis length, (29) minor axis length, (30) equivalent diameter, (31) extent, and (32) convex area.	

4.3.4 Feature ranking and selection

A common problem in most classification processes is the large number of extracted features compared to the number of observations, leading to over-fitting. There were 32 features extracted from each PF patch, some of which may be correlated, redundant or not useful. Therefore, a feature ranking and selection technique was used to reduce correlated measurements and to select the most discriminating parameters; an unsupervised filter-based feature (Infinity feature) selection method (Roffo et al., 2015b). Only 15 features were chosen as reported in Section 4.5.1.

4.3.5 Radial basis function neural network

Radial basis function neural network (RBF-NN) (Broomhead and Lowe, 1988; Moody and Darken, 1989) has been widely used in the field of pattern recognition and digital image processing due to its simplicity, functional approximation, interpolation and generalization capabilities (Borş and Pitas, 1999). The RFB-NN was designed as a three-layer feed-forward neural network topology: an input layer feeding the feature vectors into the RBF-NN; a hidden layer with radial basis function as activation function and high dimensionality structure; and an output layer where all the adjacent layer nodes were fully connected and the linear combination of the hidden weighted radial basis functions was calculated (Orr et al., 1996). In the PF segmentation process, the selected features of overlapping patches were applied to the previously trained RBF-NN as input vectors in order to classify the PF images into PF and non-PF regions. The architecture of RBF-NN model is graphically illustrated in Figure 4.5. The output of a RBF-NN model can be calculated by the following equation:

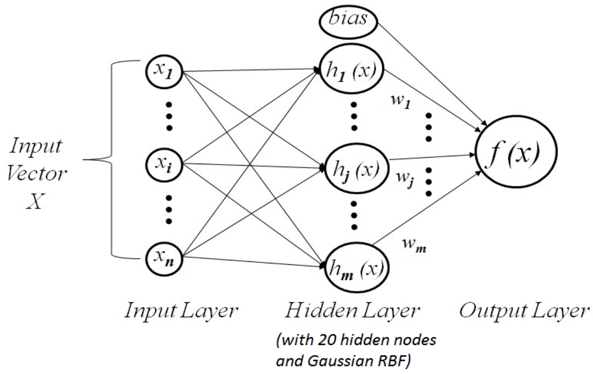


Figure 4.5: A graphical representation of RBF-NN architecture

$$f(x) = \sum_{j=1}^{m} w_j h_j(x) + B_k W_k, k = 1,2...m, \tag{4.5}$$

where $x \in R^n$ is an input feature vector, $h_j(x)$ is the basis function of the network from R^n to R, m is the number of hidden units in the hidden layer, and w_j values are weights of the network, B_k and W_k are the unit positive bias and weight from the bias neuron, respectively (an extra basis function whose output is fixed at 1 serves as the bias for each output unit). A Gaussian function (Borş and Pitas, 1996) and Stochastic gradient-based supervised learning strategy were used to update all parameters of RBF-NN, including the radial basis function centers, the widths of the Gaussian radial basis functions, and the output weights. The error cost on the input/output pattern of the RBF feed-forward neural networks can be calculated using the following equation:

$$E(x) = \frac{1}{2}\left[t(x) - f(x)\right]^2, \tag{4.6}$$

where $x \in R^n$ is an input vector, $t(x) \in \{0, 1\}$ is the corresponding target output, $f(x)$ is the actual output defined by equation (4.5).

4.3.5.1 Training and testing the RBF neural network

The RBF-NN was created as one-hidden-layer feed-forward neural network topology with 20 hidden nodes determined experimentally based on the minimum mean square error, with radial basis function as activation function, and one output layer. The proposed RBF-NN segmentation method was applied on all PF ultrasound images. The PF images were divided into small overlapping patches of size 9×9 and overlap of 4.5 pixels, where their features were extracted. The selected feature vectors were regarded as the input vectors of the RBF-NN classifier. In the classification process, the convergence conditions of the RBF-NN were set to 10^4 for maximum progress epochs and less than 10^{-5} for the correction value of synaptic weights. When one of these conditions was satisfied, the training process was terminated. The neural network model was tuned using the k-fold with 'leave-one-out' cross-validation approach where k is equal to the total number of selected features (Bishop, 2006), and the input and target vectors is automatically split into training, validation, and testing samples. For the training record, we used 60% for training, 20% for testing, and 20% for validation. A total of 300 training patterns (150 PF and 150 non-PF region textures) extracted by experienced physicians were used to train the RBF-NN. The training process continued until validation improvement was achieved. The testing data provided a separate measure of RBF-NN accuracy. 60 PF US images were used to create the testing dataset.

4.3.5.2 Segmentation of plantar fascia region using RBF

The next step was to analyse and trace the PF region of the US images using the connected component labeling algorithm presented in (Di Stefano and Bulgarelli, 1999; Gonzalez et al., 2010). This algorithm is used to assign or divide each PF component based on the image boundaries function using 4 or 8-connectivity; 8-connected neighbourhood connectivity was used to trace and label the PF region. The largest connected components extracted from the classified PF US image were considered as a part of the PF area. In the labeling algorithm process, seed equivalences were processed directly in the initial scan so that classes sharing the same set of intensity values were always sorted and updated at once during the first scan. This is maintained by assigning a unique new label to each new equivalence class and merging the corresponding classes as soon as a new equivalence is determined.

4.3.6 PF thickness measurement and estimation

The PF thickness estimation process is summarized in the following methods: a) Thick 1 method: (1) distance transformation was applied to the segmented PF US image using Euclidean distance metric (Shih, 2009), so that all background pixels were set to 0 and all foreground pixels were set to the distance from the background; (2) the local maxima pixel set points (spot centers) of the distance transformed segmented PF image were found (i.e. distances from the background). These local maxima points are also known as skeleton centered points (ridges) (Blum, 1967) with respect to the shape boundary (Telea, 2014); and (3) the thickness was computed as the median of the local maxima pixel set points. b) Thick 2 method: For each PF US segment, we computed the following parameters using property measures of the PF region and morphological operations (Ganzalez and Woods, 2002): (1) the area as the total number of pixels in the PF binary image region; (2) the length of the PF using morphological operations on the PF binary region, such as remove and skeleton; and (3) the mean PF thickness was computed as PF area divided by PF length.

The calculated PF thickness using Thick 1 and Thick 2 methods was compared against manual clinical measurements to assess the performance of the developed methods. All thickness measurements were reported in millimetres (mm). All the images used were 3 cm deep, which translated to a conversion of 1 cm for 156 pixels.

4.3.7 Alternative methods used for performance comparisons

Due to the absence of different automatic segmentation methods in the PF US imaging field, the proposed method was compared with only two different region based active contour segmentation methods: (1) semi-automated active contour model (snakes) by Kass et al. (1988), incorporating different active contour energy factors; and (2) automated localizing region based active contour method by Lankton and Tannenbaum (2008). Both methods are based on the variational information of grayscale intensities of the image, and they performed poorly when there was no much difference between the foreground and background means, especially in PF US images. In order to increase the variation information of the grayscale intensities in the US images, we introduced two different stages to tune and initialize the parameters of the images: (1) preprocessing was performed using (a) contrast limited adaptive histogram equalization (CLAHE), (b) manual and automatic initial contour mask initialization; and (c) definition of the number of iterations; (2) applying morphological operations (Ganzalez and Woods, 2002), such as (a) opening, closing, thresholding, in order to remove falsely identified small segments (usually due to image speckle noise and small variation in image intensities), (b) region filling and labeling, where the final segmented area was filled and labeled. For the first method (semi-automatic), the initial contour mask was defined manually by selecting a random set of points near PF ROIs which were later interpolated into a contour. The iteration number was set experimentally to 100, and images were resized to $1/2$ the original size for fast computation. In the second method, the active contour was automatically initialized using a predefined initial mask (4-element vector) for different PF US images and the iteration number was set experimentally to 800.

4.4 Performance evaluation protocol

4.4.1 Classification evaluation

Different performance metrics were used to evaluate the performance of the RBF-NN classifier: accuracy, true negative rate (TNR) (Sokolova and Lapalme, 2009), and cross-entropy error (CE) (Rubinstein, 1997). These measures are defined as follows:

$$Accuracy = \frac{TP+TN}{N}, \tag{4.7}$$

$$TNR = \frac{TN}{TN+FP}, \tag{4.8}$$

$$CE(X,Y) = -\frac{1}{N}\sum_{i=1}^{n} y^{(i)}ln(a(x^{(i)}) + (1-y^{(i)})ln(1-a(x^{(i)})), \tag{4.9}$$

where TP and TN represent true positive and true negative values, respectively, calculated from a confusion matrix, N is the total number of all values in the confusion matrix classes including: TP, TN, FP (false positive), and FN (false negative). TNR represents specificity (the probability of the correctly classified non-positive elements as predicted negative). $X = x^{(1)},\ldots,x^{(n)}$ is the set of input selected features in the training dataset, and $Y = y^{(1)},\ldots,y^{(n)}$ is the set of corresponding labels for input features. The $a(x)$ represents the output of the neural network for the given input feature set x.

4.4.2 Segmentation evaluation

Two different quantitative evaluation metrics found in the literature were considered to evaluate the segmentation method including region-based metrics (area overlap measures) (Udupa et al., 2006) and distance based metrics (Heimann et al., 2009). Their mathematical representations are summarized below:

4.4.2.1 Region based metrics

Region based performance metrics (Udupa et al., 2006; Rueda et al., 2014) are used to calculate precision, Dice similarity (Dice, 1945) and accuracy (using sensitivity and specificity) of the proposed segmentation method. These metrics are defined as follows:

$$Precision = \frac{\left|S_r \cap R_{gt}\right|}{\left|S_r \cup R_{gt}\right|}, \tag{4.10}$$

$$Dice = \frac{2\left|S_r \cap R_{gt}\right|}{\left|S_r + R_{gt}\right|}, \tag{4.11}$$

$$Sensitivity = \frac{\left|S_r \cap R_{gt}\right|}{\left|R_{gt}\right|}, \tag{4.12}$$

$$Specificity = \frac{\left|S_r \cup R_{gt}\right|}{\left|R_{gt}\right|}, \tag{4.13}$$

where S_r denotes the segmented results, R_{gt} represents the reference ground truth image defined by experts, $|.|$ denotes the magnitude, $\cap$ denotes the intersection (the number of common pixels in both segmented results and ground truth), and $\cup$ is the union (the number of all ground truth pixels defined by expert and the segmented results).

4.4.2.2 Distance based metrics

Different distance-based metrics (Heimann et al., 2009; Rueda et al., 2014) were applied including Hausdorff and mean sum of square distance (MSSD) metrics. These metrics are defined as follows:

1) Hausdorff distance, also known as maximum symmetric contour distance (MSD), is defined as:

$$MSD\left(R_{gt}, S_r\right) = \max\left(\left(D_1\left(C_{R_{gt}}, C_{S_r}\right), D_1\left(C_{S_r}, C_{R_{gt}}\right)\right), \tag{4.14}$$

where $C_{R_{gt}}$ and C_{S_r} denote the reference ground truth contour and segmented result contour of R_{gt} and S_r, respectively. $D_1\left(C_{R_{gt}}, C_{S_r}\right)$ and $D_1\left(C_{S_r}, C_{R_{gt}}\right)$ can be calculated using Euclidean distance as follows:

$$D_1\left(C_{R_{gt}}, C_{S_r}\right) = \max_{x_1 \in C_{R_{gt}}}\left(\min_{x_2 \in C_{S_r}}\left(\|x_1 - x_2\|\right)\right) \tag{4.15}$$

$$D_1\left(C_{S_r}, C_{R_{gt}}\right) = \max_{x_2 \in C_{S_r}}\left(\min_{x_1 \in C_{R_{gt}}}\left(\|x_2 - x_1\|\right)\right) \tag{4.16}$$

where x_1, x_2 denote the contour elements of C_{S_r} and R_{gt}, respectively, and $\|.\|$ represents the Euclidean distance.

2)The MSSD is defined by:

$$MSSD\left(R_{gt}, S_r\right) = \frac{1}{N} \sum_{n=1}^{N} D_2^2\left(C_{R_{gt}}, C_{S_r}\left(x_n\right)\right), \tag{4.17}$$

where N denotes the size of the segmented result contour, $C_{R_{gt}}$ and C_{S_r} represent the reference ground truth contour and segmented result contour of R_{gt} and S_r, respectively. $D_2\left(C_{R_{gt}}, C_{S_r}\right)$ can be calculated using:

$$D_2\left(C_{R_{gt}}, C_{S_r}(x)\right) = \min_{y \in R_{gt}}\left(\|y - x\|\right) \tag{4.18}$$

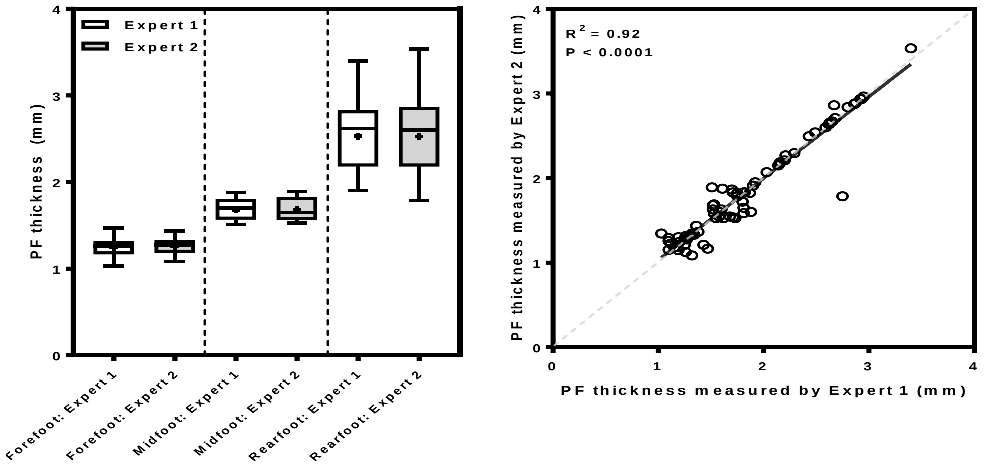
where x, y denote the contour elements of C_{S_r} and $C_{R_{gt}}$, respectively, and $\|.\|$ represents the Euclidean distance.

4.4.3 Establishing the ground truth inter-operator variability

Two medical experts, with different levels of experience (3-5 years), performed independent manual segmentation of the plantar fascia region (Figure 4.4) and measured the thickness independently using each image. The datasets generated by the two experts were used to establish the ground truth values of the plantar fascia region thickness. Intra- and inter-operator variability was assessed using several metrics as presented in Table 4.2, with the two operators presenting very close results for all segmentation metrics used. Inter-operator variability of the PF thickness measurements was also assessed using a t-test and linear regression analysis, as reported in Table 4.2, indicating consistent reproducibility (Figure 4.6).

Table 4.2: Intra- and inter-operator variability of manual segmentation of PF structure

Intra-operator differences	Region Based Metrics					Distance Based Metrics	
Metrics	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)	Dice (%)	Hausdorff (mm)	MSSD (mm)
Operator 1	98.08 ± 2.07	97.87 ± 1.07	95.97 ± 1.45	99.10 ± 1.18	96.65 ± 1.60	2.26 ± 1.62	0.66 ± 0.81
Operator 2	98.01 ± 2.00	97.65 ± 1.80	95.14 ± 1.95	98.94 ± 1.54	96.35 ± 1.69	2.41 ± 1.80	0.42 ± 0.56
Inter-operator differences							
Metrics	Accuracy (%)	Precision (%)	Sensitivity (%)	Specificity (%)	Dice (%)	Hausdorff (mm)	MSSD (mm)
Operator 1 *vs* Operator 2	98.06 ± 1.81	97.77 ± 1.25	95.73 ± 1.62	98.87 ± 1.34	96.89 ± 2.61	2.78 ± 1.56	0.74 ± 0.64
Linear regression analysis	R^2	0.92	P	<0.0001			
Paired *t*-test	P	0.853					



(a)

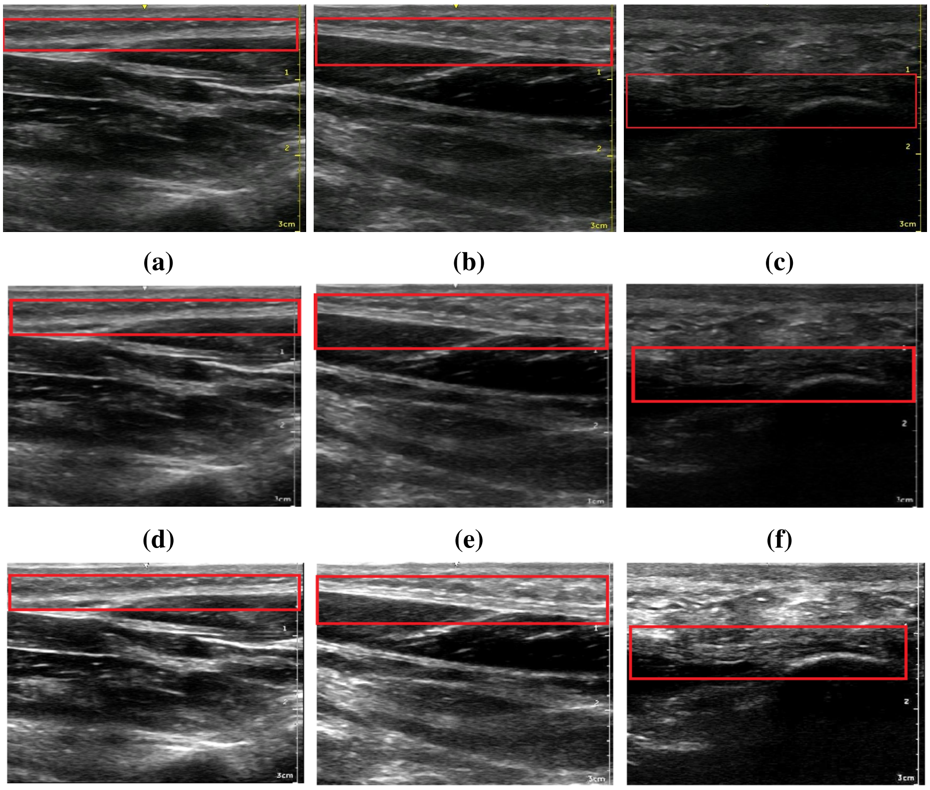
(b)

Figure 4.6: Inter-operator variability: (a) differences in thickness measurements of PF structures performed by the two experts, indicating lack of differences. The boxes show the 25th and the 75th percentiles, the whiskers denote the minimum and maximum values, the bars represent the medians, the + sign represents the means. (b) Linear regression of measurements performed by the two experts, indicating consistent pairing. The dashed line represents the line of unity and the continuous line represents the line of regression ($R^2 = 0.92$)

4.4.4 Statistical comparison between manual and automatic segmentation

Three different statistical tests were performed to assess the validity of automatic segmentation methods in relation to manual measurements, including multiple regression analysis, repeated ANOVA test and post-hoc paired *t*-test in order to analyse the pairing between the PF thickness taken manually and the estimation methods, and to demonstrate that PF thickness varies along the sites of measurement. The alpha value for statistical significance was set at 0.025 based on a Bonferroni correction. All the statistical analyses were computed using GraphPad Prism Software version 7.01 (GraphPad Software, CA, USA).

4.5 Experimental results and discussion



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.7: Preprocessing results: (a)-(c) Original US images for different PF structures (Forefoot, Mid and Rear section). (d)-(f) Speckle reduction results using DT-CWT filter (reduces noise and improves the visual quality of the image). (g)-(e) Enhancement results using CLAHE filter (PF region has been enhanced and well defined)

Different experiments were performed to prove the capability of the proposed supervised ANN segmentation method including the preprocessing stage. Figure 4.7 shows the results of applying the preprocessing methods using DT-CWT and CLAHE filters for despeckling and enhancement operations.

4.5.1 Feature selection and classification results

Feature selection analysis results of the 15 highest ranked predictors computed from 150 PF US images are shown in Table 4.3 and Figure 4.8. For each feature, the weight predictor was computed and the features were assigned a rank order according to their predictor weights. The reason for feature ranking and selection analysis is to determine the best discriminatory features that define PF area and to eliminate similar or highly co-dependant features. It is clearly evident, from feature selection analysis results (Table 4.3 and Figure 4.8), that the best 15 features were the ones with the highest ranked predictors (based on their importance weights) and which gave the best classification results. The main features included contrast, sum average, sum of squares and difference variance. The best result for RBF-NN classifier was also achieved with the best 15 selected feature set compared to other sets. The percentage of correctly classified PF segments was 98.80%. The performance measures of the RBF-NN classifier are shown in Table 4.4. The best result for RBF-NN classifier was achieved with the selected 15 feature set with a high mean accuracy of 98.75%, high mean TNR of 99.37% and low mean CE of 0.0182.

Table 4.3: Feature selection analysis results of the best 15 extracted features

Feature No.	Selected Feature Sets	Predictor Importance Weight	Predictor Rank Order
Haar Wavelet			
1	HaarVariance	7.549	5
2	HaarMean	5.789	15
Gray Level Difference Statistics (GLDS)			
10	Contrast	6.081	10
11	Correlation	6.038	11
12	Energy	5.802	14
13	Homogeneity	6.195	7
Spatial Gray Level Dependence Matrices (SGLDM)			
15	Contrast	23.775	1
16	Correlation	6.116	9
17	SumofSquares	12.225	3
19	InvDiffMoment	5.892	12
20	SumAverage	17.426	2
23	DifferenceVariance	9.092	4
Region Based Features			
27	Area	5.831	13
28	Perimeter	6.293	6
29	MajorAxisLength	6.136	8

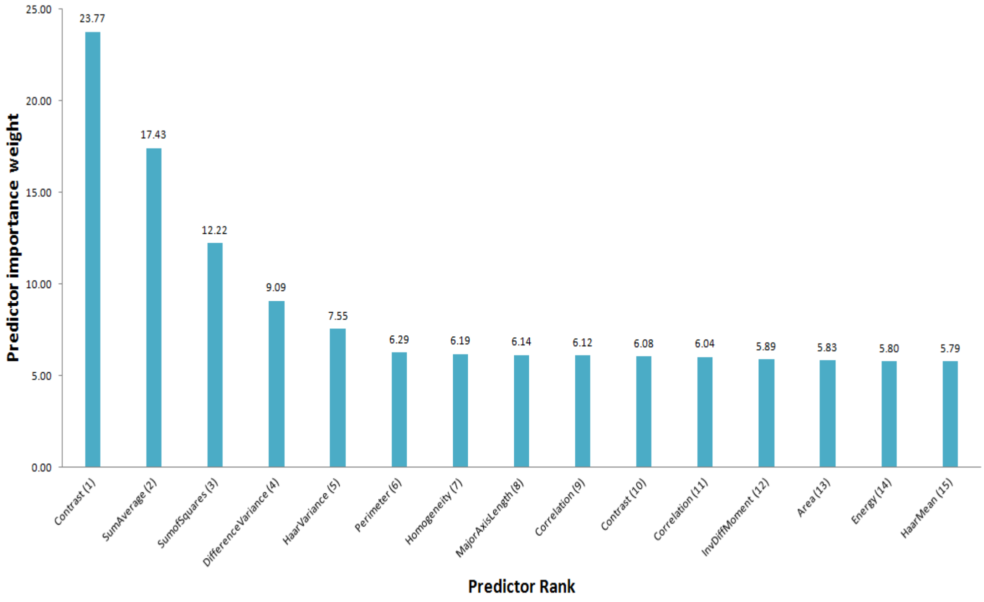


Figure 4.8: A bar plot of ranked predictors (features importance) based on importance weights

Table 4.4: The performance measures of the RBF-NN classifier using different selected feature sets

Feature sets	Accuracy	TNR	CE
Best (5) Selected Features	98.735	99.3650	0.115
Best (10) Selected Features	98.736	99.3660	0.090
Best (15) Selected Features	**98.751**	**99.3720**	**0.018**
Best (20) selected features	98.750	99.3717	0.058
All (32) features	98.741	99.3700	0.083

4.5.2 Segmentation results

The segmentation results are shown in Figures 4.9, 4.10, and 4.11, and tabulated in Tables 4.5 and 4.6. Figure 4.9 shows the segmented PF region results outlined manually by a physician (red contours) for different PF sites (Forefoot, Midfoor and Rearfoot sections), the segmentation results of the proposed method (green contours) using the RBF-NN classifier, and the binary mask of segmented PF region results. Figure 4.10 shows the segmentation results (binary mask of segmented PF region) of the semi-automatic region based active contour (snakes) method (Kass et al., 1988), and Figure 4.11 shows the segmentation results of localizing region based active contour (fully-automated) method (Lankton and Tannenbaum, 2008), as described previously. The segmentation results shown in Figures 4.9, 4.10, and 4.11 demonstrate that the proposed method correctly and precisely segments the PF area in all different structures compared to the selected active contour based methods.

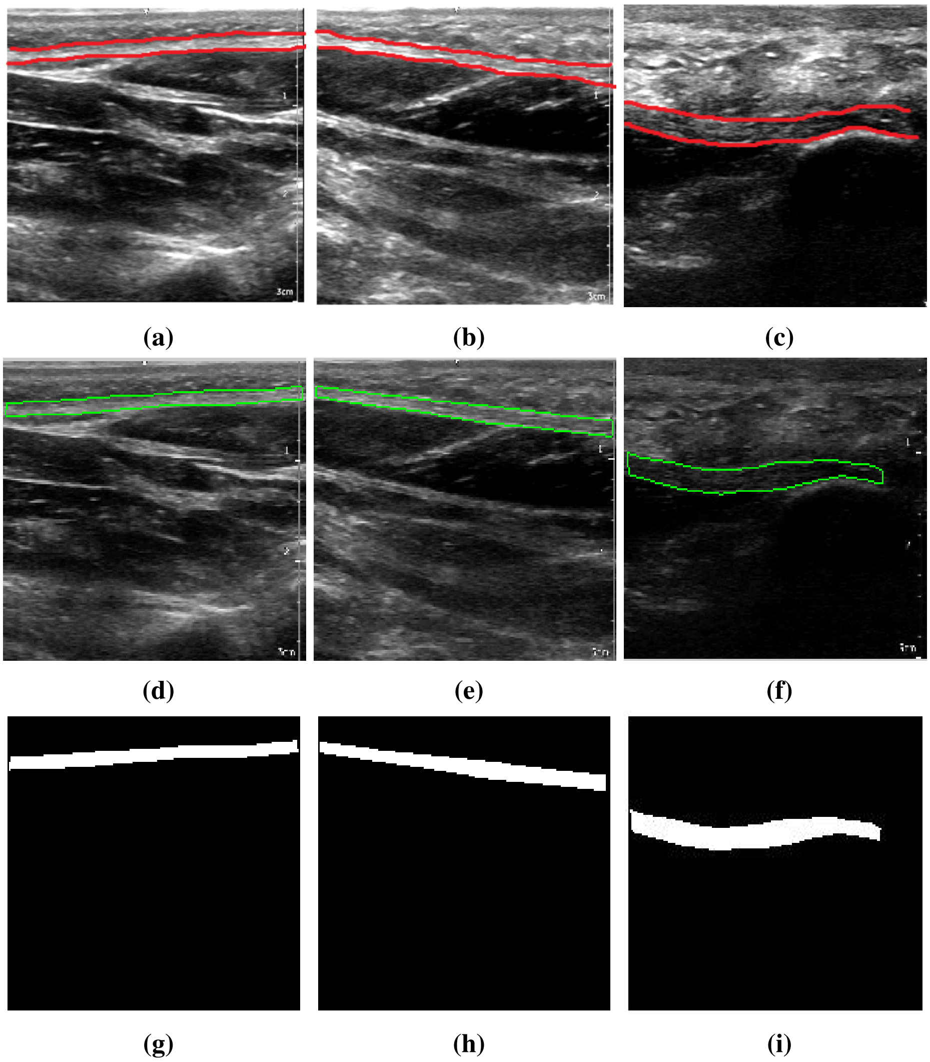
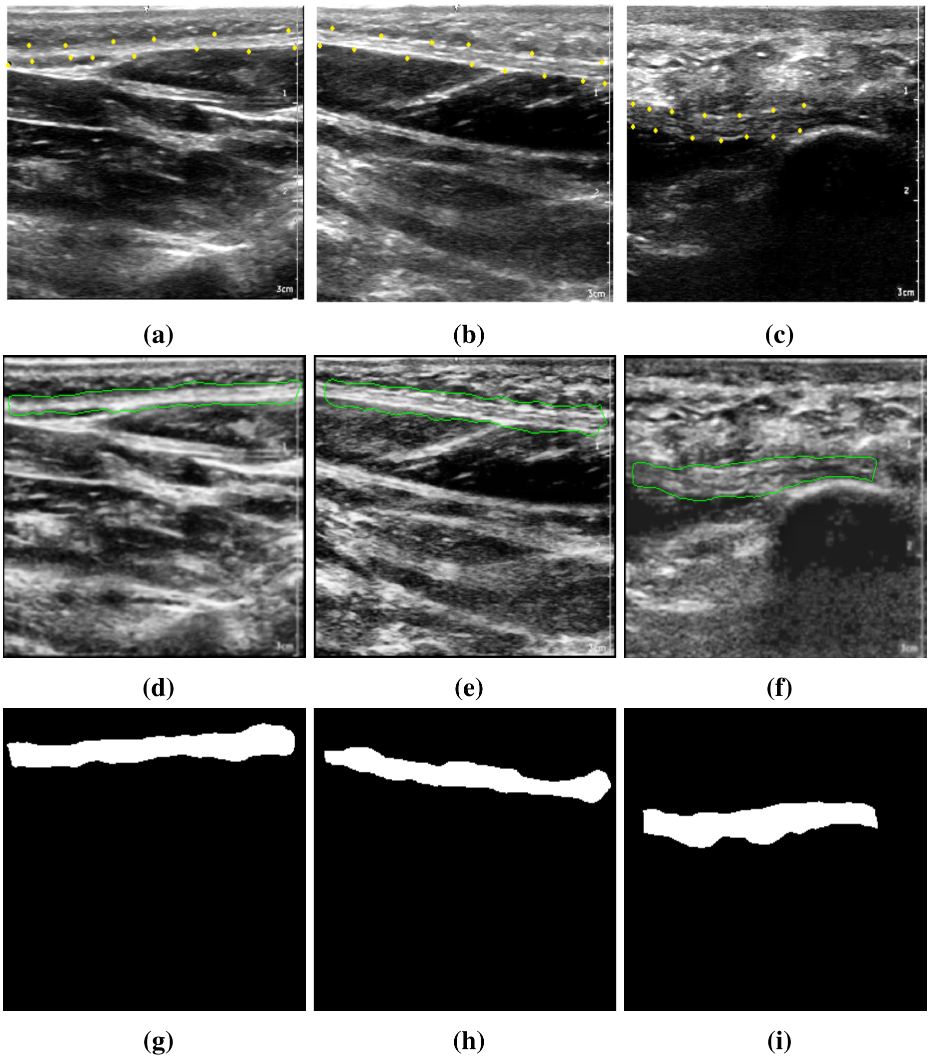


Figure 4.9: Segmentation results of the proposed method. (a)-(c) PF region outlined manually by a physician (red contours). (d)-(f) Segmented PF region result produced by RBF-NN classifier (green contours). (g)-(i) Binary mask of segmented PF region results produced by RBF classifier

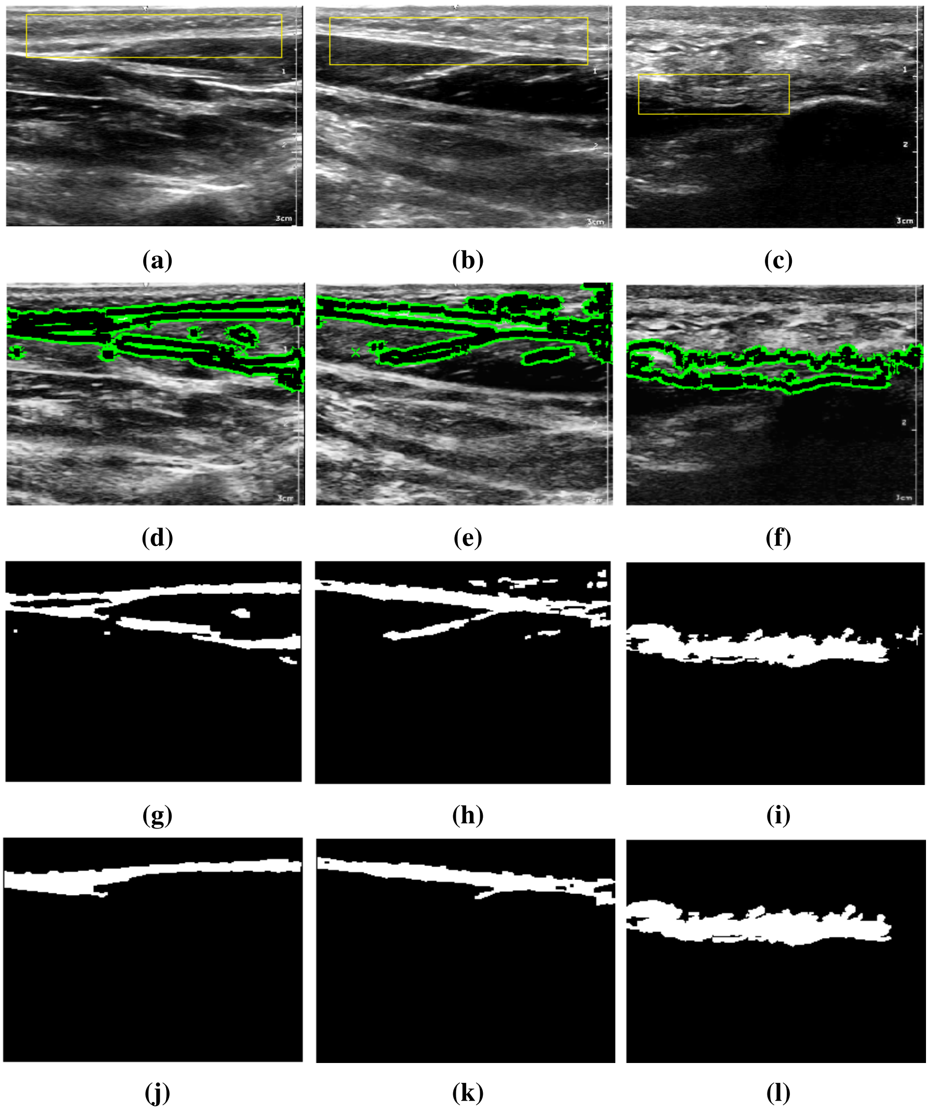


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Figure 4.10: Segmentation results of a semi-automatic region based active contour (snakes) method. (a)-(c) Active contour initialization using a manual snake mask initialization (red dots). (d)-(f) Preliminary active contour segmentation results (green contours).(g)-(i) Final selected PF region



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

(j) (k) (l)

Figure 4.11: Segmentation results of a fully automatic localizing region based active contour method. (a)-(d) Predefining the initial mask. (d)-(f) Active contour initialization using a predefined initial mask. (g)-(i) Preliminary region based segmentation results. (j)-(l) Final results using morphological operations such as: opening, closing, thresholding, and region filling

Table 4.5 summarizes the quantitative segmentation results (mean ± STD) of the proposed method. The results of six different PF US image sets (25 US images per set) show that all obtained values were close to the ground truth values for both region based metrics and distance metrics (reported only in the PF US images where there were no failures). The ranges of overall means were 96%-98% for precision, 96%-99% for

Dice similarity, 98%-100% for sensitivity, 97%-99% for specificity, 1 mm-2 mm for Hausdorff and 0.1 mm-0.2 mm for MSSD, with low standard deviation values, indicating better segmentation in terms of precision, Dice similarity, sensitivity and specificity (closer to 100% in similarity criteria), and low Hausdorff and MSSD values (closer to zero in similarity criteria) in the range 0 mm-15 mm and 0 mm-5 mm, respectively for effective segmentation results.

Table 4.5: Quantitative segmentation evaluation of the proposed method (Mean ± STD)

US Images	Region-Based Metrics				Distance-Based Metrics	
	Precision (%)	Dice (%)	Sensitivity (%)	Specificity (%)	Hausdorff (mm)	MSSD (mm)
Image_Set 1	96.98 ± 1.43	97.69 ± 1.72	98.50 ± 1.05	98.90 ± 1.06	1.79 ± 1.66	0.11 ± 0.05
Image_Set 2	97.70 ± 1.33	97.78 ± 1.93	99.34 ± 0.66	98.79 ± 1.21	1.78 ± 0.57	0.20 ± 0.21
Image_Set 3	97.13 ± 1.87	97.29 ± 1.71	98.90 ± 1.10	97.99 ± 1.01	1.57 ± 0.42	0.31 ± 0.37
Image_Set 4	96.41 ± 1.75	96.61 ± 1.60	98.62 ± 1.06	97.49 ± 1.93	1.32 ± 0.25	0.10 ± 0.07
Image_Set 5	97.00 ± 1.13	97.81 ± 0.82	98.76 ± 0.58	98.48 ± 0.72	1.56 ± 0.60	0.28 ± 0.36
Image_Set 6	97.70 ± 1.20	98.43 ± 0.96	98.61 ± 1.43	98.43 ± 1.08	1.61 ± 0.89	0.22 ± 0.29
Mean ± STD	97.15 ± 0.49	97.60 ± 0.61	98.77 ± 0.32	98.38 ± 0.54	1.60 ± 0.17	0.20 ± 0.10

Tables 4.6 compares the performance of the proposed method with the selected active contour based methods. From these results, it is evident that the proposed method exhibits the best performance in terms of all calculated segmentation metrics, with high mean values equal to 97.15% for precision, 97.60% for Dice similarity, 98.77% for sensitivity and 98.38% for specificity, and low mean values equal to 1.6 mm for Hausdorff distance and 0.2 mm for MSSD, with low standard deviation values. These results demonstrate the effectiveness of the proposed segmentation approach, with advantages over other methods. The active contour methods used in the evaluation and testing relied on variation information of edge intensity, and in most cases they are likely to miss certain parts of the PF region during the active contour segmentation process, leading to false segmentation of other objects in the image with similar intensity values to the PF region. This would require more processing steps, including despeckling, enhancement and morphological operations, such as opening, closing and filling (as shown in Figures 4.10, and 4.11).

Table 4.6: Segmentation performance metrics of different segmentation methods and the proposed method (Mean ± STD)

Segmentation Methods	Region-Based Metrics				Distance-Based Metrics	
	Precision (%)	Dice (%)	Sensitivity (%)	Specificity (%)	Hausdorff (mm)	MSSD (mm)
Lankton & Tannenbaum	76.73 ± 11.02	78.02 ± 9.89	75.67 ± 9.97	91.81 ± 5.35	3.00 ± 2.59	1.31 ± 0.30
Kass et al.	92.28 ± 7.14	81.21 ± 2.95	73.00 ± 5.93	96.58 ± 3.95	2.26 ± 0.18	1.45 ± 0.50
Proposed Method	97.15 ± 0.49	97.60 ± 0.61	98.77 ± 0.32	98.38 ± 0.54	1.60 ± 0.17	0.20 ± 0.10

4.5.3 Thickness estimation results

Table 4.7 shows the automatic thickness estimation results using the two proposed methods (Thick 1 and Thick 2) for all different PF structures compared with manual measurements (established in Subsection 3.3). To demonstrate the significant positive relationship between the manual thickness measurements and the two automatic methods and that the thickness of the PF varies along its length, regression analysis, ANOVA and post-hoc *t*-test statistics were carried out. The results are presented in Tables 4.8 and 4.9. The regression analysis presented in Table 4.8 revealed a significant positive pairing between manual measurement and the automatic Thick 2 method for all PF measurement sites ($R^2 = 0.995$, $R^2 = 0.952$ and $R^2 = 0.960$ for rearfoot, midfoot and forefoot, respectively, $p < 0.0001$). However, there was no significant positive relationship between manual measurement and Thick 1 method ($p > 0.05$). The results of the ANOVA test presented in Table 4.9 distinguished between different PF sites using the three different methods; $F = 189.5$ was obtained for manual measurements, and 159.7 for Thick 2, showing comparable results. All three approaches distinguished between the different sites, $p < 0.0001$. The repeated paired post-hoc *t*-test revealed a significant difference between different PF sites in all thickness calculation methods (manual, Thick 1 and Thick 2), $p < 0.0001$. The results presented here show that thickness values obtained using the second automatic method (Thick 2) were closer to the ground truth thickness values than the first method (Thick 1). In addition, the statistical analysis showed that the PF thickness varies along its length. The overall estimated PF thickness using the second method ranged from approximately 1.94 mm-3.56 mm for the rearfoot section, 1.57 mm-2.01 mm for the midfoot section, and 1.11 mm-1.57 mm for the forefoot section. Thus, the proposed method is advantageous and reliable in quantifying PF thickness in different structures anatomically located in the rearfoot, midfoot and forefoot sections.

Table 4.7: Thickness estimation by the proposed method for all different PF structures (Rearfoot, Midfoot and Forefoot sections)

Patients	Rearfoot PF Section			Midfoot PF Section			Forefoot PF Section		
	Manual (mm)	Thick 1 (mm)	Thick 2 (mm)	Manual (mm)	Thick 1 (mm)	Thick 2 (mm)	Manual (mm)	Thick 1 (mm)	Thick 2 (mm)
Case 1	2.49 ± 0.030	2.82 ± 0.005	2.58 ± 0.011	1.55 ± 0.011	1.92 ± 0.002	1.67 ± 0.030	1.10 ± 0.010	1.14 ± 0.050	1.14 ± 0.019
Case 2	2.67 ± 0.033	3.07 ± 0.002	2.78 ± 0.030	1.61 ± 0.014	2.60 ± 0.033	1.73 ± 0.020	1.12 ± 0.010	1.64 ± 0.008	1.13 ± 0.007
Case 3	2.80 ± 0.040	2.99 ± 0.031	2.87 ± 0.020	1.87 ± 0.084	2.34 ± 0.001	2.01 ± 0.040	1.25 ± 0.030	1.54 ± 0.016	1.31 ± 0.004
Case 4	2.15 ± 0.020	2.38 ± 0.051	2.21 ± 0.010	1.81 ± 0.091	2.02 ± 0.004	1.87 ± 0.010	1.19 ± 0.050	1.76 ± 0.010	1.29 ± 0.023
Case 5	2.03 ± 0.040	3.00 ± 0.053	2.13 ± 0.030	1.58 ± 0.041	2.13 ± 0.022	1.63 ± 0.011	1.34 ± 0.040	1.51 ± 0.008	1.44 ± 0.011
Case 6	2.63 ± 0.040	2.24 ± 0.016	2.75 ± 0.040	1.62 ± 0.092	1.95 ± 0.005	1.69 ± 0.022	1.32 ± 0.040	1.67 ± 0.016	1.34 ± 0.001
Case 7	2.95 ± 0.027	2.47 ± 0.009	3.11 ± 0.012	1.80 ± 0.029	1.73 ± 0.034	1.94 ± 0.027	1.20 ± 0.025	1.51 ± 0.009	1.25 ± 0.014
Case 8	2.93 ± 0.015	2.41 ± 0.019	3.09 ± 0.008	1.75 ± 0.015	1.95 ± 0.002	1.90 ± 0.029	1.43 ± 0.016	1.30 ± 0.001	1.54 ± 0.007
Case 9	2.29 ± 0.041	2.36 ± 0.007	2.33 ± 0.033	1.51 ± 0.033	2.09 ± 0.018	1.57 ± 0.001	1.32 ± 0.040	1.95 ± 0.094	1.33 ± 0.004
Case 10	1.90 ± 0.035	3.17 ± 0.021	1.94 ± 0.002	1.71 ± 0.034	2.15 ± 0.009	1.76 ± 0.008	1.26 ± 0.020	1.55 ± 0.015	1.37 ± 0.022
Case 11	2.21 ± 0.036	2.50 ± 0.033	2.23 ± 0.019	1.81 ± 0.030	1.73 ± 0.012	1.89 ± 0.005	1.19 ± 0.040	1.49 ± 0.025	1.22 ± 0.004
Case 12	2.16 ± 0.045	2.82 ± 0.056	2.19 ± 0.013	1.88 ± 0.031	1.61 ± 0.030	1.96 ± 0.025	1.32 ± 0.017	1.58 ± 0.055	1.34 ± 0.011
Case 13	2.20 ± 0.041	2.86 ± 0.033	2.23 ± 0.039	1.53 ± 0.034	1.76 ± 0.009	1.57 ± 0.021	1.26 ± 0.041	1.76 ± 0.010	1.27 ± 0.005
Case 14	2.86 ± 0.042	2.82 ± 0.021	3.07 ± 0.043	1.70 ± 0.014	1.61 ± 0.031	1.89 ± 0.017	1.36 ± 0.050	1.58 ± 0.054	1.52 ± 0.014
Case 15	2.75 ± 0.035	3.19 ± 0.020	2.95 ± 0.031	1.73 ± 0.024	2.05 ± 0.012	1.90 ± 0.023	1.47 ± 0.032	1.50 ± 0.008	1.57 ± 0.009
Case 16	2.65 ± 0.065	3.74 ± 0.003	2.69 ± 0.043	1.68 ± 0.051	2.80 ± 0.081	1.74 ± 0.018	1.10 ± 0.020	2.34 ± 0.011	1.12 ± 0.001
Case 17	2.68 ± 0.052	2.22 ± 0.009	2.77 ± 0.061	1.58 ± 0.063	1.95 ± 0.001	1.64 ± 0.014	1.03 ± 0.080	1.39 ± 0.008	1.11 ± 0.001
Case 18	2.87 ± 0.043	2.66 ± 0.055	2.94 ± 0.005	1.71 ± 0.033	2.02 ± 0.008	1.76 ± 0.013	1.10 ± 0.010	1.38 ± 0.009	1.13 ± 0.005
Case 19	3.40 ± 0.044	2.46 ± 0.025	3.56 ± 0.014	1.81 ± 0.094	1.78 ± 0.008	1.92 ± 0.021	1.38 ± 0.050	1.56 ± 0.017	1.44 ± 0.023
Case 20	1.92 ± 0.025	3.17 ± 0.010	1.99 ± 0.007	1.52 ± 0.022	1.96 ± 0.002	1.59 ± 0.009	1.15 ± 0.014	1.38 ± 0.009	1.18 ± 0.006
Case 21	2.14 ± 0.042	2.04 ± 0.033	2.17 ± 0.041	1.53 ± 0.011	1.84 ± 0.001	1.57 ± 0.001	1.27 ± 0.044	1.64 ± 0.022	1.31 ± 0.003
Case 22	2.43 ± 0.033	2.82 ± 0.056	2.50 ± 0.050	1.52 ± 0.021	1.77 ± 0.006	1.62 ± 0.005	1.30 ± 0.040	1.29 ± 0.015	1.38 ± 0.009
Case 23	2.92 ± 0.035	2.86 ± 0.022	2.99 ± 0.002	1.82 ± 0.033	1.82 ± 0.004	1.90 ± 0.006	1.25 ± 0.020	1.69 ± 0.002	1.28 ± 0.005
Case 24	2.62 ± 0.025	2.50 ± 0.021	2.67 ± 0.028	1.60 ± 0.014	1.73 ± 0.011	1.77 ± 0.15	1.27 ± 0.044	1.49 ± 0.025	1.32 ± 0.013
Case 25	2.59 ± 0.021	3.17 ± 0.008	2.65 ± 0.027	1.75 ± 0.007	2.15 ± 0.009	1.88 ± 0.18	1.18 ± 0.017	1.55 ± 0.015	1.26 ± 0.006
Range	1.90 - 3.40	2.04 - 3.74	1.94 - 3.56	1.51 - 1.88	1.61 - 2.80	1.57 - 2.01	1.03 - 1.47	1.14 - 1.76	1.11 - 1.57
RMSE	—	1.11	**0.43**	—	1.49	**0.49**	—	1.60	**0.28**

Table 4.8: Regression analysis between manual and automatic assessment of PF thickness

Different Sites of PF	Rearfoot		Midfoot		Forefoot	
Thickness Calculation Methods	Thick 1	Thick 2	Thick 1	Thick 2	Thick 1	Thick 2
Multiple Regression (R^2)	0.1116	0.9947	0.06648	0.9516	0.03744	0.9603
P-Value	0.2976	<0.0001	0.3761	<0.0001	0.4295	<0.0001
Significant pairing with manual?	No	Yes	No	Yes	No	Yes

Table 4.9: ANOVA analysis of differences between manual and automatic measurements

Thickness Estimation Method	Manual	Thick 1	Thick 2
F	189.5	93.26	159.7
P-Value	<0.0001	<0.0001	<0.0001
Difference between sites?	Yes	Yes	Yes

4.6 Conclusion

It can be argued that while general methods for US image segmentation are lacking, segmentation processes tend to be application dependent. In the field of PF segmentation, no automatic segmentation methods have been developed to date, and this makes the task of segmenting PF US images in the clinic more challenging; however, this also presents an opportunity for developing novel methods to facilitate this task for clinicians. The method presented in this study used a segmentation approach with feature extraction, ranking, selection analysis and RBF-NN classifier to automatically segment the PF area and estimate its thickness. The proposed segmentation method obtained favourable results compared to other active contour methods reported in the literature. Performance evaluation showed that the proposed automatic method can successfully segment the PF region and estimate the PF thickness from US images. Such a segmentation application is not only able to significantly reduce the time required by physicians for PF pathology diagnosis, but can also reduce the subjectivity that accompanies manual delineations and thickness measurements, further assisting pathologists by facilitating early diagnosis. It is evident from the statistical analysis that the second method (Thick 2) outperforms the first method (Thick 1) in terms of significant positive pairing between the manual and automatic assessment. In addition, the findings indicate a significant difference between PF structures, strongly suggesting that the thickness of the PF varies along the length of the foot. The effectiveness of the proposed method supports the potential of its use in US imaging. However, further investigation is required to (i) evaluate the performance of the proposed model in a larger dataset of normal and abnormal US images against measurements by a larger number of experts, (ii) analyse its impact on medical applications by using different ultrasound machines with advanced specifications, and (iii) improve the segmentation process such that it can classify different PF US images into normal, medium and abnormal sets.

Chapter 5

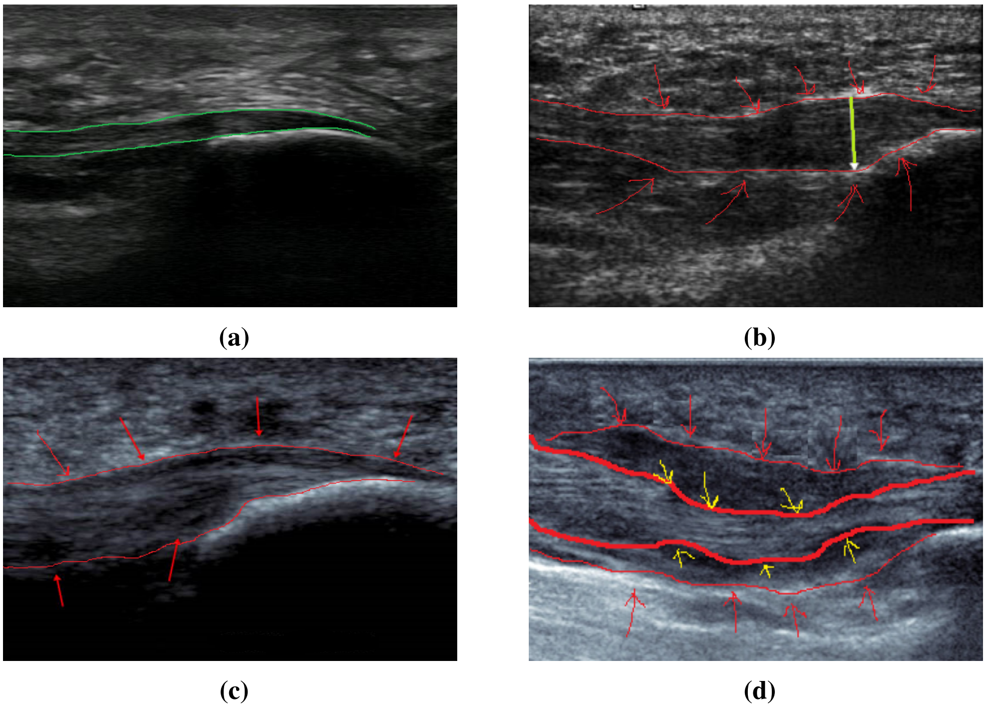
Plantar Fascia Characterization and Classification Based on Machine Learning Techniques for Ultrasound Images

5.1 Overview

This chapter introduces a supervised classification approach which distinguishes between symptomatic and asymptomatic US PF subjects. This will facilitate the characterization of the plantar fascia area for the identification of patients with inferior heel pain at risk of plantar fasciitis. This approach makes use of the following modules: (1) image despeckling and enhancement to reduce the speckle noise without loosing important information and to improve the contrast of the acquired images; (2) plantar fascia ROI segmentation using an implemented ANNs supervised segmentation approach (discussed in Chapter 4); (3) feature extraction to extract 6 different feature sets (a total of 40 features) from the segmented PF region using the following measures: spatial gray level dependence matrices (SGLDM), region based features, neighbourhood gray tone difference matrix (NGTDM), first order statistics (FOS), statistical feature matrix (SFM) and laws texture energy measures (TEM); (4) feature normalization and scaling; (5) features ranking and selection to reduce features' redundancy; (6) classification model using Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN in order to differentiate between normal and abnormal plantar fascia subjects; and (7) classification performance analysis which analyse the classification models based on three different performance analyses: (i) confusion matrix for computing recall, specificity, balanced accuracy, precision, f-score and Mtthew's correlation coefficient (MCC) measures; (ii) receiver operating characteristic (ROC) curve and area under curve (AUC) analysis; and (iii) computation complexity analysis (analysing the mathematical complexity of the selected classifiers and computing their execution time).

5.2 Introduction

As discussed earlier in the literature review US imaging offers significant potential in diagnosis of plantar fascia (PF) injuries and monitoring treatments. In particular US imaging has been shown to be reliable in foot and ankle assessment and offers a real-time effective imaging technique that is able to reliably confirm structural changes, such as thickening, rupture and identify changes in the internal echo structure associated with diseased or damaged tissues. PF US images are usually examined and analysed by physicians radiologist based on visual perceptions and some manual biometric measurements (e.g. thickness estimation) of the PF region to identify the presence of any kind of lesions and abnormalities such as plantar faschiitis (inflammation of the plantar fascia). As reported in the literature, thickening, bi-convexity, rough surface, heterogeneous texture, decreased echogenicity, loss edge sharpness and hypoechoic deformities of the PF are considered as part of the diagnostic criteria and characteristic features of symptomatic PF; whereas surface smoothness, texture homogeneity and uniform hyperechogenicity are characteristics of asymptomatic PF subjects (Park et al., 2014; Fabrikant and Park, 2011; Wearing et al., 2007; Saber et al., 2012). Figure 5.1 shows asymptomatic and symptomatic ultrasound images with PF region outlined (red contour), as well as a thickened PF area (b and c red contours) compared to a normal PF region in (a), (d) shows the irregular outline and disrupted PF region fibres (inner red bold contour) with a surrounding fluid collection due to inflammation (outer contour). This clearly shows, the convex shape of the affected PF area, the hypoechoic changes, degeneration and loss of the organization of the internal PF structure. The red contours and arrows in (b, c and d) indicate the margins of the swollen PF region.



(a) (b)

(c) (d)

Figure 5.1: Asymptomatic and Symptomatic PF region comparison: (a) Asymptomatic PF region (green contours), (b-d) Symptomatic PF region: (b) and (c) a thickened PF sections (red arrows) compared to a normal PF in (a) due to planar fasciitis disorder, (d) a huge partial tear of the PF region: the outer red contour clearly shows a surrounding inflammation (plantar fasciitis), while the inner contour (bold red) shows the irregular outline and disrupted PF region fibres.

Despite the advantages of US imaging, the acquired images interpretation and analysis are time consuming and prohibitively expensive after a long period. This is mainly due to the large number of patients, the big medical data history accumulated in the DICOM systems and the large number of physicians required for the analysis and interpretation. The exploration of such massive medical data requires highly efficient and sophisticated techniques capable in finding the class separation between asymptomatic and symptomatic ultrasound images of the plantar fascia. These techniques are highly required to classify different PF US images into normal and abnormal subjects and to prune the huge accumulated data and take in consideration only the symptomatic data with the possibility of plantar fasciitis or other disorders. Therefore, it is a requirement to devise an automated system to characterize and classify PF US images that allows better abnormalities detection and easier interpretation during medical analysis.

This study proposes a supervised classification approach which for the first time facilitate the detection and the characterization of the plantar fascia region for the classification of PF US images dataset into symptomatic PF subjects and asymptomatic subjects; and the possibility of the identification of patients with normal plantar fascia but at risk of plantar fasciitis disorder. The developed system applies the following: preprocessing, PF segmentation, feature extraction, ranking and discriminative feature selection, features characterization and analysis; PF US images classification using different classifiers modules such as Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN, and classification performance evaluation.

5.3 Proposed plantar fascia classification model

The proposed PF classification model consists of the following modules as illustrated in Figure 5.2: (1) preprocessing phase employing speckle noise reduction filtering and image enhancement operations to reduce the effects of undesirable speckle noise phenomenon and improve the contrast of the PF US images using dual tree complex wavelet transform with soft thresholding (DT-CWT_S) and contrast-limited adaptive histogram equalization filter (CLAHE), respectively; (2) artificial neural networks supervised segmentation phase applying different features measures, features ranking module and trained radial basic function neural network (RBF-NN) classifier as discussed earlier in Chapter 4 and in (Boussouar et al., 2017a) to automatically segment the PF region and calculate its thickness using average thickness expressed as PF area divided by PF length; (3) texture features extraction and analysis introducing 6 sets of feature extraction measures (for extracting a total of 40 features), features ranking and selection operation using an unsupervised infinity feature selection method (Roffo et al., 2015b) to select and analyse the most discriminating and suitable features for the classification process; (4) classifiers modules using different classification approaches such as Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN in order to distinguish between asymptomatic and symptomatic plantar fascia subjects; and (5) classification performance analysis (to select the best model) introducing 8 different performance measures such as recall, specificity, balanced accuracy, precision, F-score, MCC, AUC and computation complexity.

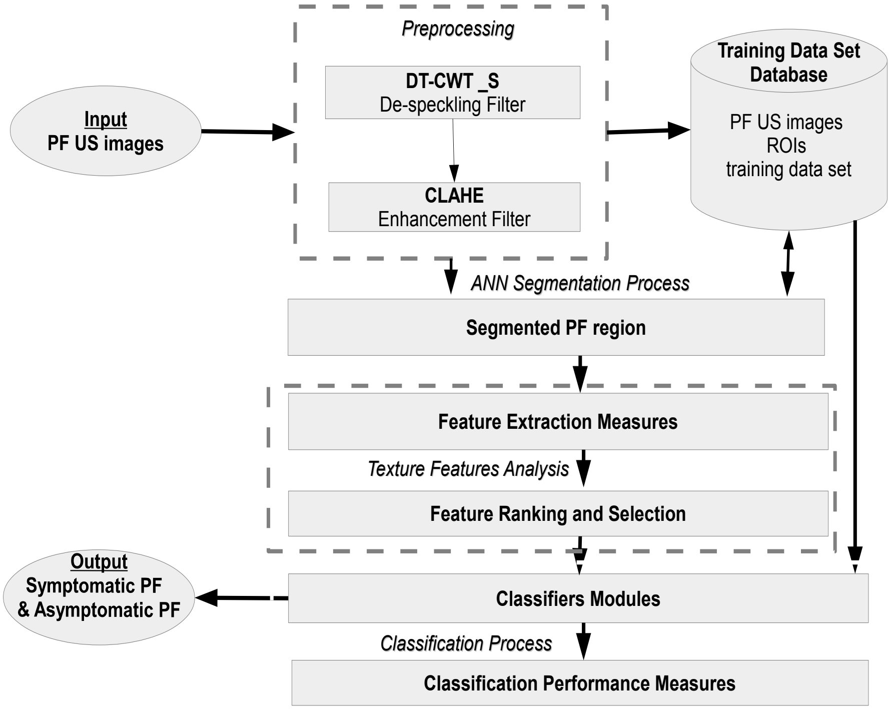


Figure 5.2: Flowchart illustrating the plantar fascia classification system based on a Texture features analysis and different classifiers modules

5.4 Methodology

5.4.1 Materials and PF US images data collection

Following ethics approval from the University of Salford Research's Ethics Panel (ST1617-48), written informed consent was collected from all patients participants. Various PF US images, acquired from a patient's footprint area in the prone position were used in the classification approach (Figure 5.3); more specifically, a total of 284 (252 normal and 32 abnormal taken from diabetic patients with plantar fasciitis) PF

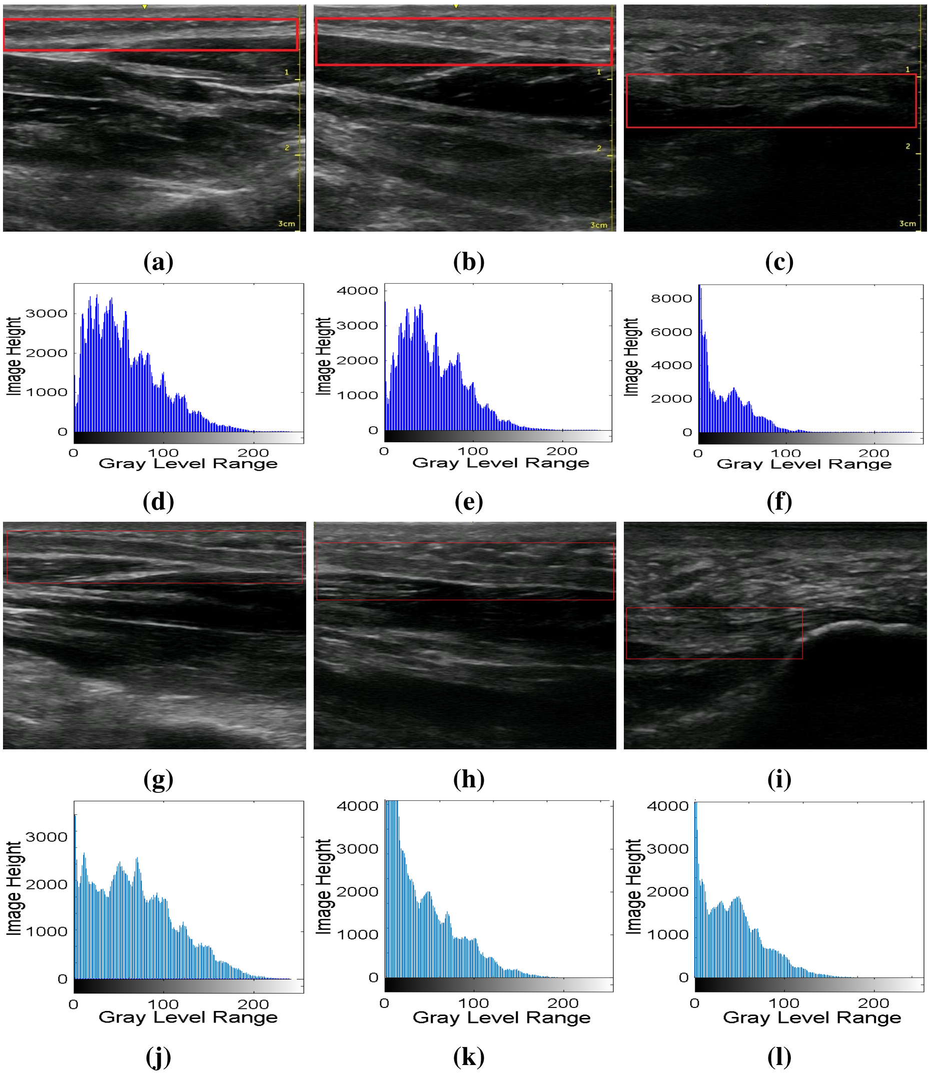


Figure 5.3: US images for different PF structures: (a)-(c) Asymptomatic PF US samples (Forefoot, Midfoot and Rearfoot sections, respectively); (d)-(f) their normal gray level histogram representation; (g)-(i) Symptomatic PF US samples (Forefoot, Midfoot and Rearfoot section, respectively); (j)-(l) their abnormal gray level histogram representation.

US images were obtained from 45 patients for different PF anatomical structures including rearfoot, midfoot and forefoot sections with 256 gray levels, a size dimension of 512×512 pixels and a resolution of 28.35 pixels/centimeter. These images were obtained from the Health Sciences Department, University of Salford, acquired by two expert clinicians according to a precise protocol using a Venue 40 musculoskeletal US system (GE Healthcare, UK) with a $5-13$ MHz wideband linear array probe

$(12.7\ mm \times 47.1\ mm)$. All the methods used in the proposed approach were implemented using Matlab R2017b (The MathWorks Inc., Natwick, USA). This will be discussed in more details in Chapter 7 (Methods implementation and development process).

5.4.2 Preprocessing

The Preprocessing phase aims: (1) to prepare the PF US images for further processing including segmentation and classification and improve their accuracy, efficiency, and scalability; (2) to minimize the effects of the multiplicative speckle noise without loosing any valuable information (such as tiny lines, edges); (3) to enhance the PF region contrast; (4) and to visually improve the global appearance of the PF US images.

5.4.2.1 Despeckling

For the last 20 years, wavelet-based despeckling approaches have been studied widely in medical imaging due to their good performance and advantageous properties such as multi-directionality, multi-resolution and multi-energy compaction (Dhawan, 2011; Kingsbury, 1998, 1999; Selesnick et al., 2005). In this classification study, we used a selected dual tree complex wavelet-based despeckling filter (DT-CWT_S) based on the previous speckle reduction evaluation approach discussed earlier in Chapter 3. This filter integrates homomorphic transformation (using log compression and exponent decompression to transform the multiplicative noise to an additive one) and multi-scale DT-CWT decomposition and composition employing the BayesShrink subband thresholding using soft thresholding to reduce or suppress the speckle noise (noisy coefficients) in PF US images. DT-CWT_S has demonstrated a superior edge preserving behaviour and a good visual appearance in our study. The following steps summarize wavelet based filtering process as described earlier in Chapters 2, 3 and 4: a) homomorphic transformation using log compression filter; b) DT-CWT image decomposition using dual tree (real and imaginary parts) complex wavelet decomposition; c) threshold estimation, modification and suppression of noisy coefficients using BayesShrink thresholding rule (Chang et al., 2000) and simple soft thresholding function (Sendur and Selesnick, 2002a); d) application of inverse DT-CWT for signal composition; and e) exponential transformation to obtain despeckled signal.

5.4.2.2 Contrast enhancement

In medical imaging applications such as segmentation and image registration, CLAHE (Zuiderveld, 1994) is advantageous in enhancement of low-contrast images when compared to standard adaptive histogram equalization (AHE) (Pizer et al., 1987); where the histogram is calculated for the contextual region of a pixel. In this study, the CLAHE enhancement was performed after despeckling operation to adjust the intensity of the PF region using different implemented steps as described previously in Section 4 and reported in (Zuiderveld, 1994) and to avoid noise amplification in PF US images.

5.4.3 Segmentation

Automated segmentation is one of the most important tasks in medical image processing and analysis, including, pattern recognition, supervised or unsupervised subjects classification and novelty detection; it is mainly used to locate the desired region of interest objects in the input images dataset. As reported in (Boussouar et al., 2017a), an automated ANNs supervised segmentation approach was introduced in this study to segment different PF regions. The proposed segmentation approach uses the radial basic function neural network (RBF-NN) classifier (Ham and Kostanic, 2000) to automatically segment the PF region and estimate its thickness. All the segmentation process steps are described in details in (Boussouar et al., 2017a) and illustrated in Fig. 5.4:

5.4.4 Feature extraction

In most classification tasks, feature extraction is an important step to extract the relevant information (reduced input dataset representation) from the input dataset in order to perform the remaining tasks. Thus, the main goal of feature extraction in this classification study is to extract a set of textual features from the PF segments (using different measures) that discriminate between one input pattern from another pattern, and then fed into different classifiers for a classification task. In this stage six different sets of features (40 features) were extracted from the segmented PF region including, (i) haralick spatial gray level dependence matrices (SGLDM) (Haralick et al., 1973), (ii) Region based features, (iii) Neighbourhood gray tone difference matrix (NGTDM) (Amadasun and King, 1989), (iv) Histogram based features or first-order statistics (FOS) (Umbaugh,

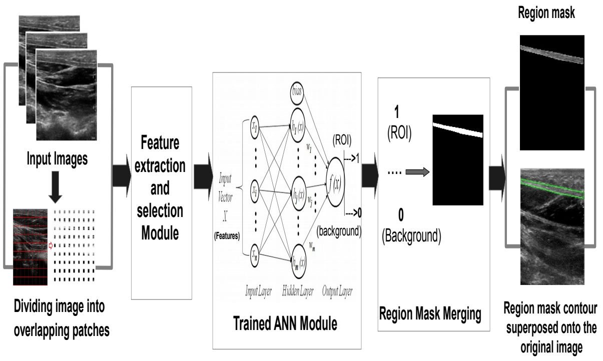


Figure 5.4: Block diagram showing ANNs approach for segmenting PF ROIs from PF US images dataset: (1) input dataset (preprocessed PF US input images with overlapping patches); (2) features extraction and selection; (3) PF image patches classification (PF and non-PF) using trained RBF-ANN classifier; (4) region mask formation(black background and white PF ROI); (5) region mask labelling and superposing.

2005; Christodoulou et al., 2003), (v) Statistical feature matrix (SFM) (Wu and Chen, 1992; Christodoulou et al., 2003), (vi) Laws' texture energy measures (TEM) (Wu et al., 1992; Laws, 1980; Christodoulou et al., 2003) All the feature sets used in this study were also successfully introduced in some previous texture analysis related studies (Weszka et al., 1976b; Ojala et al., 1996; Christodoulou et al., 2003). All the features extracted may have some redundancy, thus we need to introduce feature selection and analysis stage to reduce this redundancy and to select the most discriminant feature sets.These features were extracted using the following measures:

5.4.4.1 Haralick spatial gray level dependence matrices

Spatial Gray Level Dependence Matrices (SGLDM) is reported as the most popular statistical feature extraction method. It is firts proposed by Haralick (1973), and it takes in consideration the spatial relationship between pixels in the image. In this study, the following 12 SGLDM features were computed and averaged for a selected distance $d = 1$ (3×3 matrices) and four different orientation angles $\theta = 0^o, 45^o, 90^o$, and 135^o: (1) angular second moment, (2) contrast, (3) correlation, (4) sum of squares, (5) variance, (6) inverse difference moment (InvDiffMoment), (7) sum average, (8) sum variance , (9) sum entropy, (10) entropy, (11) difference variance, and (12) difference entropy.

5.4.4.2 Region based features

Region based features also called shape based features which are mainly related to shape, size and orientation of the ROI in the image. The following seven shape based features were computed from the PF segments: (1) area, (2) perimeter, (3) major axis length, (4) equivalent diameter, (5) extent, (6) convex area, and (7) orientation.

5.4.4.3 Neighbourhood Gray Tone Difference Matrix

Neighbourhood gray tone difference matrix (NGTDM) was first proposed by Amadasun and King (1989) to extract the visual properties of the image (or the segmented ROI area) texture. In this study, the following five NGTDM features were extracted from the PF segments for a kernel window (neighborhood) size of 3x3: (1) Coarseness, (2) contrast, (3) busyness, (4) complexity, and (5) strength.

5.4.4.4 Histogram features

Histogram based features also known as first-order statistics (FOS) (Umbaugh, 2005; Christodoulou et al., 2003) measures the gray level distribution texture characteristics of an image or a sub-image using the histogram representation and the gray levels distribution. The following eight FOS features were computed from the segmented PF regions: (1) mean, (2) variance, (3) skewness, (4) kurtosis, (5) energy, and (6)entropy.

5.4.4.5 Statistical Feature Matrix

Statistical feature matrix (SFM) (Wu and Chen, 1992) computes the statistical characteristics of two pixel sets at various spaces within an image. For SFM algorithm, the following four features were calculated: (1) coarseness, (2) contrast, (3) periodicity, and (4) roughness.

5.4.4.6 Laws Texture Energy Measures

For extracting Laws' texture energy measures (TEM) (Wu et al., 1992; Laws, 1980; Christodoulou et al., 2003) three vectors (L,E and S) of size 5 were employed as reported in (Wu et al., 1992), including L=(1,4,6,4,1); E=(-1,-2,0,2,1); and S=(-1,0,2,0,-1). Where L carry out local averaging, E operates as an edge detector filter and S as spot detector. The 5x5 laws masks were first defined by multiplying column vectors by row vectors (with the same length 5), then these masks are convoluted with original image and its energy statistics were used to calculate the following 6 texture energy features: (1) LL derived from LL kernel window; (2) EE derived from EE kernel window; (3) SS derived from SS kernel window; (4) average LE derived from LE and EL kernel windows; (5) average ES derived from ES and SE kernel windows; and (6) average LS derived from LS and SL windows.

5.4.5 Feature normalization

The mean variance normalization (MVN) approach is used in this study to normalize PF feature vectors in such a way that normalized PF feature vectors are more suited for all classification models. MVN is also known as zero-mean and unit-variance normalization method which helps in reducing any non-linear distortion and scaling all features so they fall within a specified range (e.g $[0\ 1]$ or $[-1\ 1]$). It transforms the feature vector to a random variable with a mean value of zero and variance of one (Dougherty, 2012). As we are dealing with some feature extraction methods involving area, perimeter, diameter and distance measurements, features normalization would also prevent these measurement values from outweighing other feature values with smaller values.

Suppose $X_j = \{x_{j,1}, x_{j,2}, ..., x_{j,n}\}$ is our input feature vector data, where $x_{j,n}$ is the n^{th} feature of the i^{th} block of size M^2. The normalized features '$NX_{j,n}$' are computed by calculating the difference between the features and their mean values, and then divided by their standard deviation values as given by equation 5.1 (Dougherty, 2012):

$$NX_{i,n} = \frac{x_{j,n} - \mu_{j,n}}{\sigma_{j,n}}, \tag{5.1}$$

where $\mu_{j,n}$ is the mean value of the feature vector x_i and $\sigma_{j,n}$ is its standard deviation.

5.4.6 Feature ranking and selection

A common deficiency in most pattern recognition and classification tasks is the high dimension of the extracted feature space compared to the number of the input samples (40 features × 284 observations). This will lead to some common problems such as: over-fitting, poor generalization and high computation cost. In order to minimize the aforementioned problems, a combination feature ranking and feature selection unsupervised infinity techniques (Roffo et al., 2015a, 2017) were introduced to reduce the correlated measurements and to select the most discriminating features. Different selected feature sets were analysed to choose the best discriminating features for different classification modules based on high F-score values.

5.4.7 Classification

Following feature ranking and selection analysis, feature classification approach was implemented using different common classifier modules. This aims to classify the plantar fascia into symptomatic or asymptomatic subjects using six selected classifiers: (1) Linear-SVM; (2) Kernel-SVM; (3) LDA; (4) KNN; (5) CART-DT; and (6) RBF-NN) (Dhawan, 2011). In order to obtain a good classification results, three main conditions were taken into consideration during the classification process: (1) careful selection of features; (2) a good classifier; and (3) suitable training samples (Unger et al., 2015). All the analysed feature sets described earlier in Section III, were treated as input vectors to the selected classifier modules and their results were evaluated using different classification measures. In order to overcome the over-fitting problem and to validate the robustness of different classifiers, cross-validation task was also introduced using k-folded (k=10 folds) approach to randomly select the training and testing instance classes. In following subsections, we will only focus on six chosen supervised machine learning modules, starting with linear and kernel SVM.

5.4.8 Support vector machines (SVM)

Support vector machines (SVM) (Vapnik, 2013) is widely used in bioinformatic and medical studies for pattern recognition related problems (Martínez-Trinidad et al., 2006). The main concept of SVM is that, firstly, it differentiates between two class samples according to the optimal maximum margin (distance between each set) hyperplane (or decision boundary) search result (Unger et al., 2015); secondly, if the hyperplane fails to split the previous linear class samples, the SVM makes use of different kernel functions such as polynomial kernel, Gaussian-RBF and sigmoids-NN unstead of linear SVM (Cortes and Vapnik, 1995; Vapnik, 2013; Osuna et al., 1997). This aims to achieve high dimensional feature space when translating original data samples (Shi et al., 2010). In this study, both Linear-SVM and Kernel-SVM classifiers were tested and the Gaussian-RBF kernel function is used in the Kernel-SVM main function. For the PF US 2D training dataset T_S with N_L labelled instances (X_j, Y_j), where X_j denotes the feature instances and Y_j is the class label with 1 for normal and -1 for abnormal PF class, and N is the total number of samples (252 normal and 32 abnormal samples with 40 extracted features). The following steps take place when dealing with both linear or non-linear separable instance classes (Shi et al., 2010; Abe, 2010):

1. For linear separable classes, search for the SVM optimal hyperplanes with a maximized width margin $\frac{2}{||w||}$ defined as the difference (or distance) between the two hyperplanes (5.2) and (5.3), such that the condition given by (5.4) is achieved, and for each labelled instance (X_j, Y_j), a suitable classification operation can be determined by (5.6), and the weight vector W is calculated by (5.5).

$$WX_j + b = 1, \tag{5.2}$$

$$WX_j + b = -1, \tag{5.3}$$

$$WX_j + b = 0, \tag{5.4}$$

$$W = \sum_{i=1}^{m} \alpha_i Y_j SV_j, \tag{5.5}$$

$$Y_j(WX_j + b) \geqslant 1, i = 1, ..., n. \tag{5.6}$$

   where W is the normal weight vector to the hyperplane, b defines the current location of the hyperplane, m represents the total number of support vectors, α_i denotes the calculated non-negative coefficients and SV_j represents the support vectors.

2. For non-linear separable classes, the above definition has been extended using the SVM kernel functions and map data to high dimensional space in order to perform linear separation. Thus, the linear decision surface equation (5.4) is replaced by

non-linear decision surface equation (5.7) and linear weight vector computed by (5.5) is replaced by (5.10):

$$Y_i(WX_j + b) \geqslant 1 - \Delta_i, \quad i = 1, ..., n. \tag{5.7}$$

The SVM optimal separating hyperplane issue is to minimize (5.8) subject to equation (5.9).

$$\frac{W^2}{2} + C_p \sum_{i=1}^{m} \Delta_i, \tag{5.8}$$

$$Y_i(WX_j + b) \geqslant 1 - \Delta_i, \quad i = 1, ..., n, \tag{5.9}$$

$$W = \sum_{i=1}^{m} \alpha_i Y_i K(SV_i, X) + b = 0. \tag{5.10}$$

where Δ_i represents non-negative variables, C_p denotes the constant parameter (helps tuning and balancing between minimum classification errors and the maximized margins), K is the SVM Kernel function (in this study Gaussian-RBF kernel function was implemented). The first part of equation (5.9) computes the margin width between the support vectors in the hyperplane and the second parts calculates the total number of misclassified instances. The kernel (non-linear) weight vector W is computed by (5.10).

5.4.9 Linear discriminant analysis (LDA)

Linear discriminant analysis was first introduced and used in classification problems by Ronald Fisher (1936). The LDA classifier tries to maximize the class separability and to maintain the discriminatory information related to different data classes using both dependent and independent instance variables (Dey et al., 2017). In the LDA classification process two main steps occur: (1) calculate the distance margin ratio (in the range of [0 1]; 0 for maximum dispersion and 1 for no dispersion) between the predictor instances in the predefined class samples (to asses if there is a maximum dispersion of the class centroids) using the statistical Wilks' Lambda defined in (Legendre and Legendre, 2012); (2) find the linear combination of the class variables among the specified groups based on the highest calculated discriminant score (i.e. allocating the unknown (or new) class instance to the class samples with the highest discriminant score) (Dey et al., 2017; Legendre and Legendre, 2012). The discriminant score can be computed using equation (5.11).

$$Ds_j = \mu_j C^{-1} f_k^T - \frac{1}{2}\mu_j C_{\mu_j}^{-1} + \ln(Pp_j) \tag{5.11}$$

where Ds_j represents the discriminatory score for the j^{th} class, μ_j denotes the mean value of the feature vectors for the j^{th} class, C is the resulted variance, f_k represents the feature vector of the k^{th} unknown instance and Pp is the predefined probability for the j^{th} class. The strength of the LDA classifier is that it can successfully be used for both multi-class classification task and small dataset samples (Dey et al., 2017).

5.4.10 The K-Nearest Neighbor (K-NN) Technique

The k-nearest neighbour algorithm (k-NN) (Fix and Hodges Jr, 1951) is also considered as a supervised classification approach that uses predefined labelled classes of training examples for classifying objects in various categories depending on the nearest training samples in the feature space (according to the predefined relative k distances between known stored vectors and new or unknown vectors) (Dhawan, 2011). Assuming that N is the number of instances in the training data sets declared by N_i; $i = 1,2,3......N$, the k-NN classification process can be implemented using the following steps (Unger et al., 2015; Megalooikonomou et al., 2007):

1. The training phase; where the feature vectors and predefined class labels of the training examples are stored (no learning is performed here, i.e. lazy learning),

2. The classification phase; where the stored features are computed for the test sample (or query, whose class is not labelled) by:

- computing the relative distances between the new (unknown) feature vector and the previous known vectors using the Euclidean similarity distance measure as in (5.12) and selecting k nearest samples (the number of neighbours).

$$ECsim_i(v) = \|v - m_i\|. \tag{5.12}$$

$$m_i = \frac{1}{N_i'} \sum v_i \in N_i v_i; \quad i = 1,2,3....N, \tag{5.13}$$

where $ECsim_i$ is the Euclidean similarity distance measure, v denotes the unknown instance vector, m_i represents the mean of the instance vectors for a specific class N_i and computed by equation (5.12), and N_i' is the size of the instances vectors in the instance class N_i.

- predicting (labelling) the new point to fit in the most numerous classes within the training samples set closest to query point. This means allocating the unknown (new) instance vector to the predefined N_i class if the condition in (5.14) is true:

$$ECsim_i(v) = min_{i=1}^{N}[ECsim_i(v)]. \tag{5.14}$$

For the K-NN PF classification task, the number of neighbours was set to k=5 and the Euclidean similarity distance measure was used.

5.4.11 Decision trees

Different decision tree algorithms exist in the literature, including, ID3 (Iterative Dichotomiser 3), C4.5, (Quinlan, 1993), classification and regression trees (CART) (Breiman et al., 1984), and OC1 (Oblique Classifier 1) (Murthy et al., 1994, 1993). C4.5 (including updated versions such as See5 and C5.0) and CART algorithms were among the top selected methods due to their fast balanced process, and high classification performance (Lim et al., 2000). In this study, only CART-DT approach was selected because of its cross-validation strength that addresses over-fitting issues and it is considered as an alternative approach to regression analysis approach (Breiman et al., 1984;

Aggarwal, 2015). In order to build binary decision trees using a given training dataset with predefined labelled classes, the CART-DT was implemented as follows (Aggarwal, 2015; Breiman et al., 1984):

1. CART partitioning criteria using Gini index binary partitioning approach: the Gini index is biased towards a larger number of partitions. Gini index is defined using equation (5.15).

$$Gini_{index}(X_j) = \sum_{y \in Y} p_{jy}(1 - p_{jy}) = 1 - \sum_{y \in Y} p_{jy}^2, \tag{5.15}$$

$$\Delta IMPFGini_{index}(P_R) = Gini_{index}(T_s) \sum_{j \in P_R} \frac{|X_j|}{|T_s|} Gini_{index}(X_j). \tag{5.16}$$

   The probability that a randomly selected member of Xi is of class yj is where X_j is a subset of the training set T_s, y represents class label that belongs to the set of class labels Y, p_{jy} is the random selection probability, $\Delta IMPFGini_{index}$ in equation (5.16) is the impurity function of the Gini index, P_R denotes the partitioning rule and T_s represents the training dataset. In order to optimize the CART decision tree, the partitioning rule P_R that minimizes the Gini impurity function $IMPFGini_{index}$ should take place.

2. CART tree Pruning: using CART Gini index approach can lead to over-fitting and larger size which may alter the decision trees performance improvement. As a solution to this, CART integrates cost complexity pruning approach to get rid of complex branches or sub-trees data and its algorithmic bias during the trees partitioning stage and replace them with simpler branches or single nodes (and save them for the next move) and preserves or improves their classification accuracy. This process is repeated until only the root node remains. The cost or error complexity pruning method is defined by (5.17).

$$Cost_{error-compl} = \frac{E(T_s) - \sum_{L_j \in L_{T_s}} E(L_j)}{|L_{T_s}| - 1}, \tag{5.17}$$

   where $E(T_s)$ is the error rate for training the dataset T_s, L_j denotes the subsets of L_{T_s}, L_{T_s} represents subsets leaf node of T_s, and $E(L_j)$ is the error rate of subsets

L_j.

3. CART stopping condition: CART algorithm stops only when reaching the minimum node size condition. Alternatively, it tries to proceed with high-quality pruning process.

5.4.12 RBF neural networks technique

5.4.12.1 Training and testing the RBF neural network

In this study RBF-NN (already define in Chapter 4) was created as one-hidden-layer feed-forward neural network topology with 20 hidden nodes determined experimentally based on the minimum mean square error, with RBF as activation function, and one output layer. The RBF-NN classification method was applied on all PF ultrasound images. The extracted and selected feature vectors were treated as input vectors of the RBF-NN classifier. In the classification process, the convergence conditions of the RBF-NN were set to 10^4 for maximum progress epochs and less than 10^{-5} for the correction value of synaptic weights. When one of these conditions was satisfied, the training process was terminated. The neural network model was tuned using the 10-fold with 'leave-one-out' cross-validation approach (Bishop, 2006), and the input and target vectors is automatically split into training, validation, and testing samples. A total of 284 training patterns (252 normal-PF and 32 abnormal-PF segments) extracted using the proposed segmentation approach (discussed earlier in Chapter 4) to train the RBF-NN. The training process continued until validation improvement was achieved. The testing data provided a separate measure of RBF-NN accuracy.

5.4.12.2 Classification of plantar fascia US images using RBF model

As mention earlier, the RBF neural network is applied to classify the PF segments into normal or abnormal cases. In the PF classification process, the extracted and selected feature vectors are applied directly to the RBF-NN through the high dimensionality hidden layer in order to classify the PF US segments. The selected feature vectors are then regarded as the input training vectors of the RBF neural network. The training procedure of the RBF neural networks requires the training of all RBF neural network parameters including the centres of the hidden layer units, the widths of the corresponding Gaussian

RBFs, and the weights between the hidden layer and output layer. The trained RBF-NN classifies the PF segments into symptomatic PF and asymptomatic PF region.

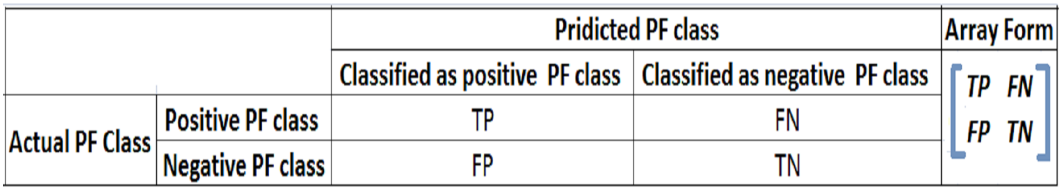
5.4.13 Classifiers Performance analysis

The correctness and the effectiveness of the different classification modules (LDA, kernel SVM, k-NN, CART decision trees and RBF-NN) in classifying symptomatic and asymptomatic PF subjects, were evaluated using the following performance measures (Dhawan, 2011):

1. Confusion matrix:

   In this study the confusion matrix (Metz, 1978) (Table 5.1) was introduced to predict the four main instances: (i) true positive (TP), the number of normal PF US subjects correctly distinguished as normal PF class; (ii) true negative (TN), the number of abnormal PF US subjects correctly distinguished as abnormal PF class with plantar fasciitis; (iii) false positive (FP), the number of normal PF US subjects incorrectly distinguished as abnormal PF class; and (iv) false negative (FN), the number of abnormal PF US subjects incorrectly distinguished as normal PF class.

Table 5.1: 2-Class Confusion Matrix predictions with actual and predicted PF classifications



		Pridicted PF class		Array Form
		Classified as positive PF class	Classified as negative PF class	$\begin{bmatrix} TP & FN \\ FP & TN \end{bmatrix}$
Actual PF Class	Positive PF class	TP	FN	
	Negative PF class	FP	TN	

From the previous four confusion matrix predictions, different classification evaluation measures (Sokolova and Lapalme, 2009; Matthews, 1975; Compton and Cao, 2006; Liu et al., 2018) were computed:

- The **Recall or Sensitivity** is the proportion of asymptomatic (positive) PF subjects that were correctly distinguished (i.e. low recall percent means the presence of high false negative predictions), as computed using (5.18).

$$Recall = Sensitivity = \frac{TP}{TP+FN} \times 100. \tag{5.18}$$

- The **Specificity or true negative rate (TNR)** is the proportion of symptomatic (negative) PF subjects that were correctly distinguished (i.e. low specificity percent indicates the presence of high false positive predictions), as defined using (5.19).

$$Specificity = TNR = \frac{TN}{FP+TN} \times 100. \tag{5.19}$$

- The **balanced accuracy (B-Accuracy)** is the mean of recall and specificity (i.e. measures the balance between the negative and positive PF predicted classes). In this study the B-Accuracy has been chosen as a better measure (it introduces all confusion matrix class attributes) than the simple accuracy, where the latter was unable to capture the whole class attributes and failed to address imbalanced dataset classification problems (Compton and Cao, 2006; Liu et al., 2018). The B-Accuracy was calculated using (5.20).

$$B\text{-}Accuracy = \frac{Recall + Specificity}{2} \times 100. \tag{5.20}$$

- The **Precision or positive predictive value (PPV)** is the proportion of the predicted asymptomatic (positive) PF classes that were correct (a low precision score indicates a high false positive instances), as computed using (5.21).

$$Precision = PPV = \frac{TP}{TP+FP} \times 100. \tag{5.21}$$

- The **F-Score** is the harmonic average of recall and precision, as calculated using (5.22) or (5.23).

$$F\text{-}Score = \frac{2TP}{2TP+FP+FN} \times 100, \;\; Or \tag{5.22}$$

$$= \frac{2(Recall * Precision)}{Recall + Precision} \times 100. \tag{5.23}$$

- The **Matthew's correlation coefficient (MCC)** (Matthews, 1975) is a binary classification (balanced) measure used in this study to calculate the correlation between the predicted PF class values and the actual PF class values using all four confusion matrix attributes (TP, TN, FP and FN). MCC ranges between -1 and 1; where -1 for a worse classification prediction, 0 for a random classification prediction and 1 for a perfect classification prediction (in this study, the percent values of MCC were calculated). MCC measure is commonly used in the literature to address the classification imbalanced dataset problems (Matthews, 1975; Măndoiu and Zelikovsky, 2007) as the case in this study. The MCC is computed using (5.24).

$$MCC = \frac{T1}{\sqrt{(T2*T3}} \times 100, \tag{5.24}$$

where $T1$, $T2$, and $T3$ were defined using (5.25), (5.26), and (5.27), respectively.

$$T1 = (TP*TN) - (FP*FN), \tag{5.25}$$

$$T2 = (TP+FP)*(TP+FN), \tag{5.26}$$

$$T3 = (TN+FP)*(TN+FN). \tag{5.27}$$

2. **Roc and AUC**:

The Receiver operating characteristic (ROC) graph analysis (Metz, 1978) is beneficial in most machine learning classification related tasks and representing their performance visually (Fawcett, 2005). In this study ROC is introduced to assess the sensitivity and specificity of the selected classification modules. ROC graph analysis plots a curve of the sensitivity (TP rate) vs. $1-$ specificity (FP rate). The best classification module will have lines heading towards the top left and the top edge line of the plotting area or close to that. The area under the curve (AUC) is also defined from the ROC graph, its values varies from 0.5 for random classification to one for perfect classification prediction. The AUC can be estimated using the trapezoid function defined by (5.28).

$$AUC = \sum_{j=1}^{j=n-1} \frac{(FP_{j+1} - FP_i) \times (TP_i + TP_{i+1})}{2}, \tag{5.28}$$

where n represents the sequence operating points, j varies from 1 to n, by assuming that the TP values are not decreased when j is increased. More details can be found in (Bradley, 1997).

3. **Average and execution cost measures**: Average measure is the mean value of Recall, Specificity, B-Accuracy, Precision, F-score and MCC measures, and it is used here to summarize all these measures. Whereas, the classification execution cost is the computation time required in predicting and using different classification models.

5.5 Experimental results and discussion

For the classification experimental results, a total of 284 (252 symptomatic and 32 asymptomatic) US images of the PF regions (rear-foot, mid-foot and fore-foot sites) were analysed. Six different sets of features (represented by shape features, intensity features, statistical and texture features) representing a total of 40 features were computed both from symptomatic and asymptomatic US images of the PF segments (segmented using the automated PF segmentation approach discussed in Chapter 3). For all extracted features, feature selection approach was introduced and their means, weights and ranking orders were computed and analysed for normal and abnormal PF US images. Figure 5.5 shows a 2-D graph plot of the top two selected features (LS v. Contrast) of 284 PF dataset (252 asymptomatic in blue and 32 symptomatic in red).

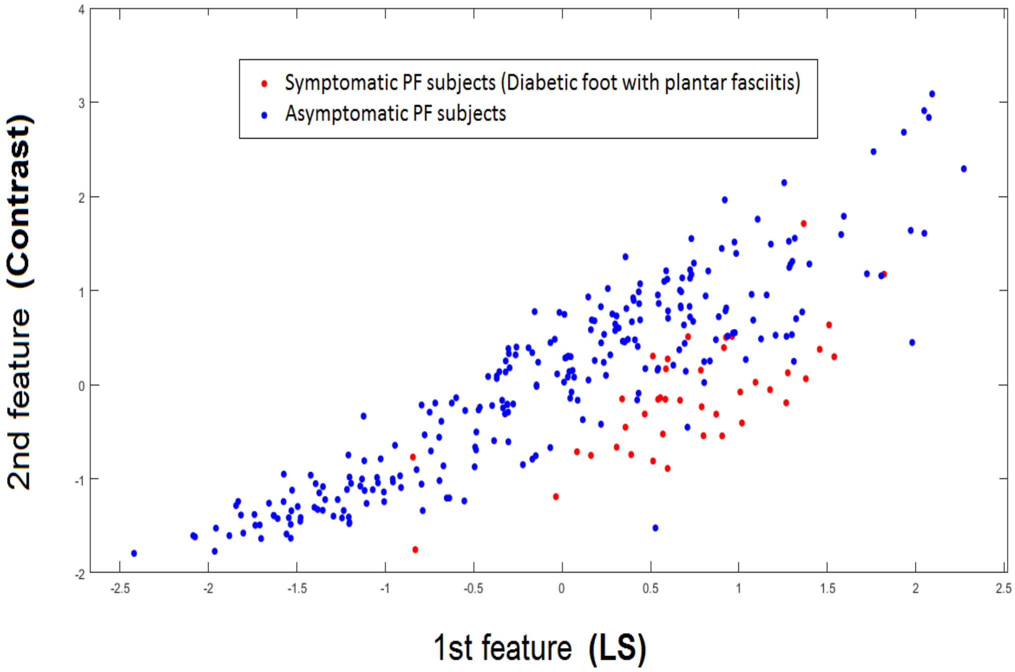


Figure 5.5: A 2-D plot representation of the top two selected features of 284 PF dataset (252 Normal and 32 Abnormal). The first feature on the x-axis is LS of Laws Texture Energy measures (LTEM); the second feature on the y-axis is the Contrast of the Histogram features. Features of normal and abnormal PF samples are shown in blue and red, respectively.

5.5.1 Feature extraction and selection analysis

The main reason for feature selection analysis in this study is to eliminate similar or highly co-dependant features and to find the best discriminatory features that predict the best classification results using different classification modules. Feature selection analysis results of the top ranked features calculated from 284 US images of the segmented PF region are represented in Table 5.2 and Figure 5.6. For each feature, the weight predictor was calculated and its rank order was assigned accordingly. Table 5.2 tabulates feature selection results of the top 34 ranked predictors based on their weights importance for asymptomatic, symptomatic PF US classes and all dataset and their rank orders were assigned. The best features (with the highest weight and ranked predictor) were found to be LS, Contrast, Variance, LE, Energy, SumSquare, AngSecMoment, LL, EE, DiffVariance, Strength, ES, Complexity, Correlation, DiffEntropy, SS, SumAverage, MajAxLength, Periodicity, Business, Mean, Skewness, Kurtosis, Orientation, Roughness, ConvexArea, Extent, EquivDiameter and Area. This has been clearly represented in Figure 5.6 where all extracted features are ranked by their weights in a descending order. Figure 5.7 shows a graphical representation of 40 feature weights computed both from symptomatic and asymptomatic US images of the PF segments. It is clearly evident from this representation, that there are differences in feature weights values between asymptomatic and symptomatic PF subjects. Table 5.3 summarizes features differences between asymptomatic and symptomatic PF subjects obtained by the interpretation of the results shown in Figure 5.7. From this interpretation, symptomatic PF texture tends to be darker with high contrast, , high variance, high shape measures (high thickness) (due to the accumulation of the inflammation fluid), more extent, high convex area (due to irregularity of the PF surface and outline disruption), high complexity (more heterogeneous), low strength, less periodicity, more roughness and low grey intensity. While on the other side, asymptomatic PF texture are brighter with low contrast, low variance, less shape measures, less extent, less convex area, low complexity (more homogeneous), high strength, more periodicity, more smoothness and high grey intensity.

Table 5.2: Feature selection analysis results of the top 34 selected features

Feature No.	Sellected Feature Sets	Predictor Important Weights			Predictor Rank Order
		Assymptomatic PF	Symptomatic PF	All Dataset	
Haralick Spatial Gray Level Dependence Matrices (SGLDM)					
1	AngSecMoment	9.85	9.90	9.82	7
2	Contrast	7.81	8.14	7.78	20
3	Correlation	9.50	10.42	9.01	14
4	SumSquare	8.60	10.18	10.09	6
5	Variance	8.77	9.14	11.04	3
6	InvDiffMoment	5.48	2.18	1.84	30
7	SumAverage	9.83	9.96	7.80	19
10	Entropy	2.94	2.96	1.79	32
11	DiffVariance	8.49	10.75	9.38	10
12	DiffEntropy	9.00	15.55	8.73	17
Neighbourhood Gray Tone Difference Matrix (NGTDM)					
13	Area	1.69	5.31	1.22	34
15	MajAxLength	6.89	7.78	7.55	21
16	EquivDiameter	1.96	5.31	1.36	33
17	Extent	1.58	4.90	1.81	31
18	ConvexArea	0.77	5.69	2.51	29
19	Orientation	5.59	4.57	3.45	27
Histogram Features					
21	Contrast	8.93	10.28	11.22	2
22	Busyness	2.16	2.35	7.23	23
23	Complexity	9.08	10.46	9.27	13
24	Strength	9.05	7.42	9.38	11
Statistical Feature Matrix (SFM)					
25	Mean	4.83	8.13	5.62	24
26	Variance	7.71	9.68	8.88	16
27	Skewness	5.57	3.15	5.47	25
28	Kurtosis	3.04	2.98	3.62	26
29	Energy	8.89	6.42	10.53	5
Statistical Feature Matrix (SFM)					
32	Contrast	10.41	10.99	8.99	15
33	Periodicity	5.87	5.63	7.51	22
34	Roughness	2.78	8.32	3.11	28
Laws Texture Energy Measures (TEM)					
35	LL	9.83	10.77	9.77	8
36	EE	9.46	10.59	9.43	9
37	SS	10.72	10.05	8.71	18
38	LE	12.49	10.85	10.68	4
39	ES	8.85	10.40	9.29	12
40	LS	10.95	10.43	11.43	1

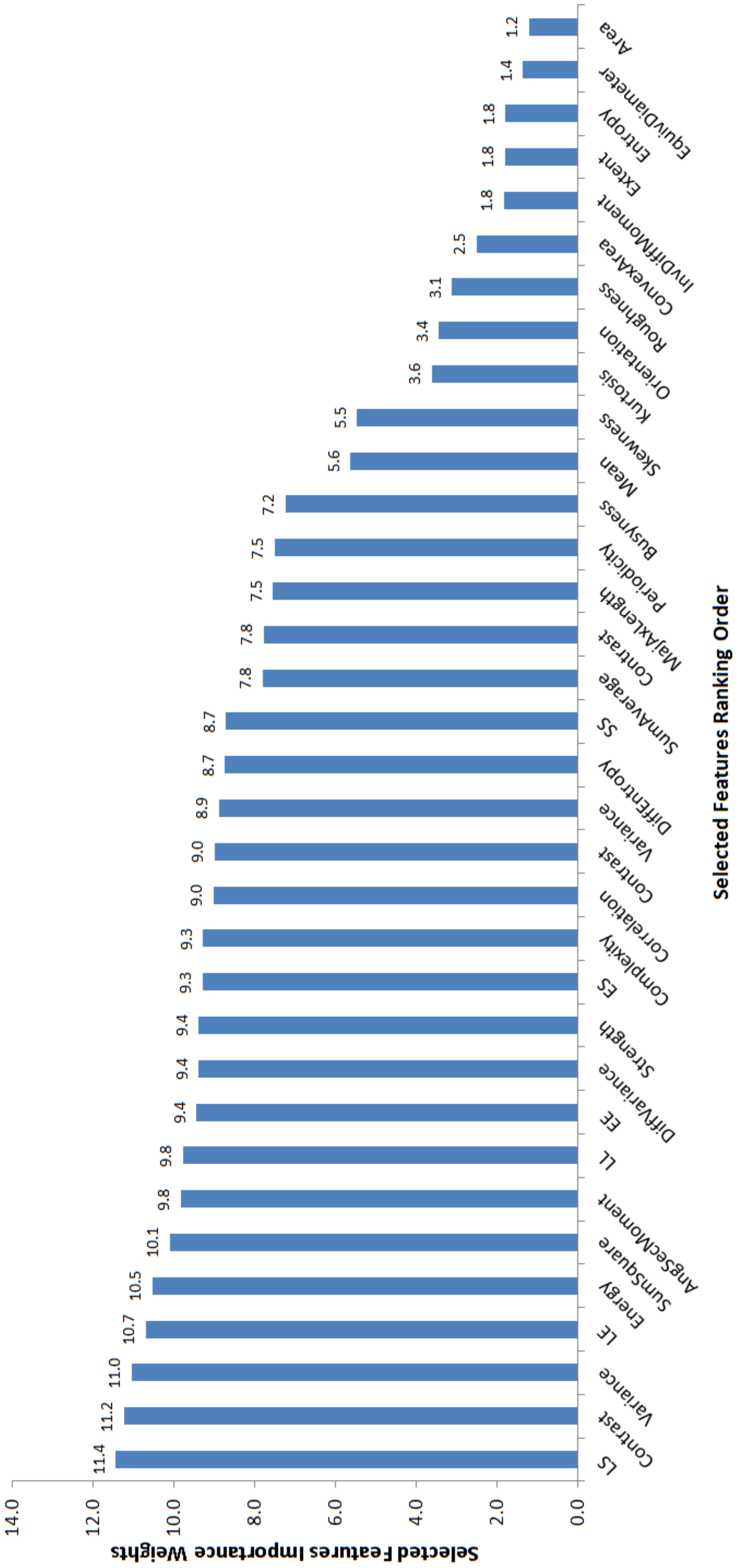


Figure 5.6: Graph representation of 34 ranked predictors (features importance) based on their importance weights

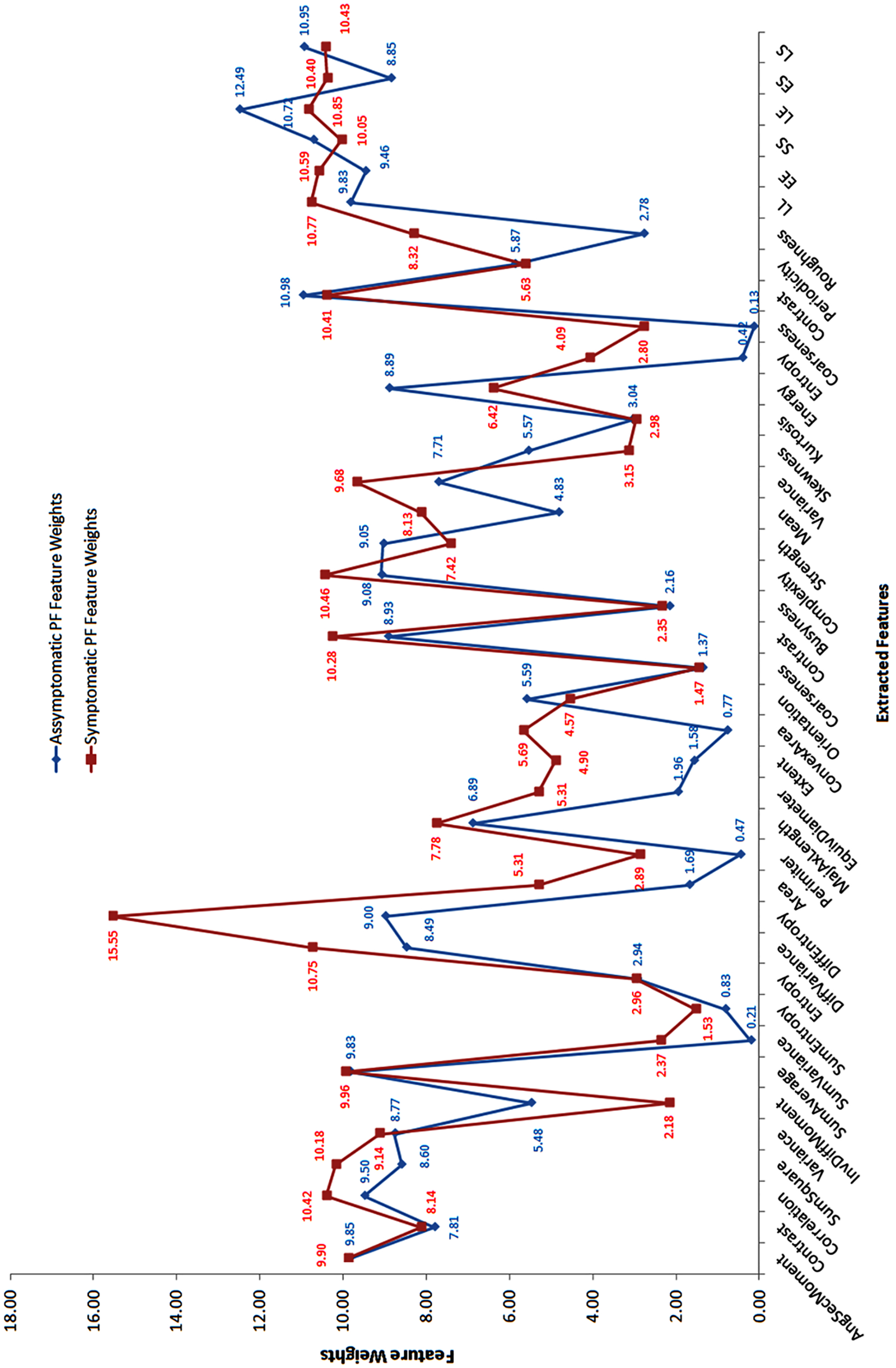


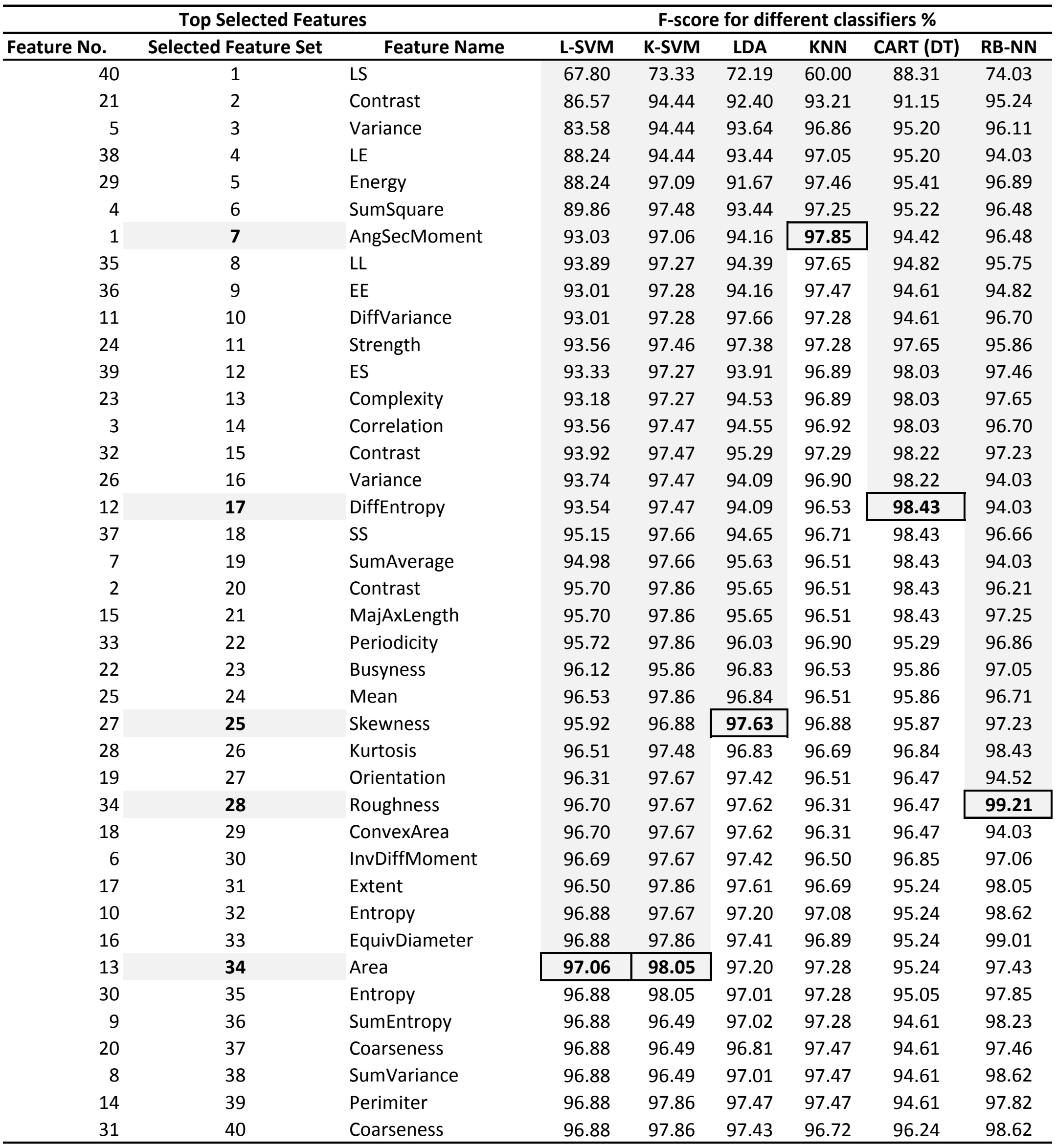
Figure 5.7: Graph representation of 40 feature weights computed both from symptomatic and asymptomatic US images (a total of 284 images) of the PF region.

Table 5.3: Asymptomatic and symptomatic PF texture characteristics interpretation resulted from Figure: 5.8

Assymptomatic PF Subjects	**Symtomatic PF Subjects**
Low contrast	High contrast
Low variance	High variance
Less area, perimeter and diameter	High area, perimeter and diameter
(Less thickness)	(High thickness)
Less extent	More extent
High orrientation	Less orientation
Less convex area	High convex area
Less complexity (homogeneous)	High complexity (heterogeneous)
High strength	Low strength
High periodicity	Less periodicity
More smoothness	More roughness
High grey intensity	Low grey intensity

In order to determine the best selected features for each classifier module, F-score measures were computed for different classifiers (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN) using different selected feature sets (from 1 to 40, starting with the highest ranked features) as presented in Table 5.4. Six selected feature sets were defined using the highest F-score measure. The best classification F-score results were achieved by RBF-NN using the selected 28-features set with a score of 99.21%, followed by the CART DT using 17-features set with 98.43%, the Kernel-SVM using 34-features set with 98.05%, the KNN using 7-features set with 97.85%, the LDA using 25-features set with 97.63% and the Linear-SVM using 34-features set with a score of 97.06%.

Table 5.4: F-score measures for different classifiers using the selected feature sets (1-40 sets) based on their weights and ranking orders.



Top Selected Features			F-score for different classifiers %					
Feature No.	Selected Feature Set	Feature Name	L-SVM	K-SVM	LDA	KNN	CART (DT)	RB-NN
40	1	LS	67.80	73.33	72.19	60.00	88.31	74.03
21	2	Contrast	86.57	94.44	92.40	93.21	91.15	95.24
5	3	Variance	83.58	94.44	93.64	96.86	95.20	96.11
38	4	LE	88.24	94.44	93.44	97.05	95.20	94.03
29	5	Energy	88.24	97.09	91.67	97.46	95.41	96.89
4	6	SumSquare	89.86	97.48	93.44	97.25	95.22	96.48
1	**7**	AngSecMoment	93.03	97.06	94.16	**97.85**	94.42	96.48
35	8	LL	93.89	97.27	94.39	97.65	94.82	95.75
36	9	EE	93.01	97.28	94.16	97.47	94.61	94.82
11	10	DiffVariance	93.01	97.28	97.66	97.28	94.61	96.70
24	11	Strength	93.56	97.46	97.38	97.28	97.65	95.86
39	12	ES	93.33	97.27	93.91	96.89	98.03	97.46
23	13	Complexity	93.18	97.27	94.53	96.89	98.03	97.65
3	14	Correlation	93.56	97.47	94.55	96.92	98.03	96.70
32	15	Contrast	93.92	97.47	95.29	97.29	98.22	97.23
26	16	Variance	93.74	97.47	94.09	96.90	98.22	94.03
12	**17**	DiffEntropy	93.54	97.47	94.09	96.53	**98.43**	94.03
37	18	SS	95.15	97.66	94.65	96.71	98.43	96.66
7	19	SumAverage	94.98	97.66	95.63	96.51	98.43	94.03
2	20	Contrast	95.70	97.86	95.65	96.51	98.43	96.21
15	21	MajAxLength	95.70	97.86	95.65	96.51	98.43	97.25
33	22	Periodicity	95.72	97.86	96.03	96.90	95.29	96.86
22	23	Busyness	96.12	95.86	96.83	96.53	95.86	97.05
25	24	Mean	96.53	97.86	96.84	96.51	95.86	96.71
27	**25**	Skewness	95.92	96.88	**97.63**	96.88	95.87	97.23
28	26	Kurtosis	96.51	97.48	96.83	96.69	96.84	98.43
19	27	Orientation	96.31	97.67	97.42	96.51	96.47	94.52
34	**28**	Roughness	96.70	97.67	97.62	96.31	96.47	**99.21**
18	29	ConvexArea	96.70	97.67	97.62	96.31	96.47	94.03
6	30	InvDiffMoment	96.69	97.67	97.42	96.50	96.85	97.06
17	31	Extent	96.50	97.86	97.61	96.69	95.24	98.05
10	32	Entropy	96.88	97.67	97.20	97.08	95.24	98.62
16	33	EquivDiameter	96.88	97.86	97.41	96.89	95.24	99.01
13	**34**	Area	**97.06**	**98.05**	97.20	97.28	95.24	97.43
30	35	Entropy	96.88	98.05	97.01	97.28	95.05	97.85
9	36	SumEntropy	96.88	96.49	97.02	97.28	94.61	98.23
20	37	Coarseness	96.88	96.49	96.81	97.47	94.61	97.46
8	38	SumVariance	96.88	96.49	97.01	97.47	94.61	98.62
14	39	Perimiter	96.88	97.86	97.47	97.47	94.61	97.82
31	40	Coarseness	96.88	97.86	97.43	96.72	96.24	98.62

5.5.2 Classification analysis

For the classification task, different classifiers modules were implemented including, Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN. All classification modules have been trained and tested using the same training and testing datasets, respectively. In order to overcome over-fitting problem during training stage and to assess the performance of various classification modules, 10-fold cross-validation was tested (Bishop, 2006). The main concept of cross-validation approach is that each sample is added in both training and testing samples. In the case of 10-fold cross validation approach, datasets (252 asymptomatic PF subjects and 32 symptomatic subjects) are randomly partitioned into 10 different equal splits (folds) (i.e. $10 - 1 = 9$ folds were used for training task and the remaining fold is used for testing, with an iteration of 10 times dropping one-fold out for testing each time). In order to select the best model for this study, 7 different classification performance measures were computed and analysed using 10-fold cross validation approach (as one evaluation measure analysis is typically not enough for selecting the best classification model). For each classification module the mean value of the 10-cross validations was computed and the results of each module are shown in Table 5.5 and Figures 5.8, 5.9, 5.10, 5.11, 5.14, 5.13, and 5.15. In general, a perfect classifier would correctly predict 100% true positive subjects (252 asymptomatic PF subjects), 100% true negative subjects (32 symptomatic PF subjects with plantar fasciitis), 0% false negative and 0% false positive attributes.

Table 5.5 tabulates the 10-fold classification performance results using the confusion matrix and different performance measures (Recall, Specificity, B-Accuracy, Precision, F-score, MCC percent, AUC and Time Cost) of different classifiers (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN) using different selected feature sets (defined earlier in the previous subsection).

Table 5.5: The performance measures of different classifiers using the best selected feature sets

Classifier	**Best Selected**	**Confusion Metric**				**Performance Measures (%)**							**ROC Plot**	**Time Cost**
Type	**Feature Set**	**TN**	**FN**	**FP**	**TP**	**Recall**	**Specificity**	**B-Accuracy**	**Precision**	**F-Score**	**MCC**	**Average**	**AUC**	**(sec)**
Linear-SVM	34	21	11	4	248	95.75	84.00	89.88	98.41	97.06	71.46	89.43	0.94	0.49
Kernel-SVM	34	22	10	0	252	96.18	**100.00**	**98.09**	**100.00**	**98.05**	**81.32**	**95.61**	**0.97**	0.36
LDA	25	26	6	6	246	**97.62**	81.25	89.43	97.62	97.63	78.87	90.40	**0.97**	**0.31**
KNN	7	23	9	2	250	96.53	92.00	94.26	**99.21**	97.85	79.33	93.20	0.94	0.34
CART DT	17	26	6	2	250	**97.66**	**92.86**	**95.26**	**99.21**	**98.43**	**85.34**	**94.79**	0.91	**0.30**
RBF-NN	28	29	3	1	251	**98.82**	**96.67**	**97.74**	**99.60**	**99.21**	**92.82**	**97.48**	**0.98**	**0.18**
							All Features							
Linear-SVM	40	20	12	4	248	95.38	83.33	89.36	98.41	96.88	69.25	88.77	**0.93**	0.65
Kernel-SVM	40	21	11	0	252	95.82	**100.00**	**97.91**	**100.00**	**97.86**	**79.30**	**95.15**	**0.95**	0.47
LDA	40	25	7	6	246	**97.23**	80.65	88.94	97.62	**97.43**	**76.80**	89.78	**0.93**	0.42
KNN	40	16	16	1	251	94.01	**94.12**	**94.06**	**99.60**	96.72	66.12	**90.77**	0.92	**0.37**
CART DT	40	22	10	9	243	**96.05**	70.97	83.51	96.43	96.24	66.09	84.88	0.83	**0.40**
RBF-NN	40	27	5	2	250	**98.04**	**93.10**	**96.48**	**99.21**	**98.62**	**87.28**	**95.46**	**0.97**	**0.21**

Figures 5.8 and 5.9 show the graphical illustration of the average confusion matrix attributes (TP, FP, TN and FN) and the misclassified instances (the sum of FP and FN instances) of different classification modules (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT, RBF-NN) using the best selected feature sets and all (40) feature sets. These attributes (TP, FP, TN and FN) were used to calculate most of the classification measures (Recall, Specificity, B-Accuracy, Precision, F-score, MCC percent, AUC) of different classifiers in differentiating between asymptomatic and symptomatic PF subjects. For the best selected features (Figure 5.8), it can be seen that the RBF-NN classifier was the best in term of low PF misclassified instances only 4 out of 284 PF instances followed by CART DT with (8/284) and Kernel-SVM with (10/284). From Figure 5.9, it is also seen that the RBF-NN classification model achieved the best result in terms of a small number of misclassified PF instances with only 7 out of 284 PF instances followed by Kernel-SVM with (11/284) and LDA with (13/284).

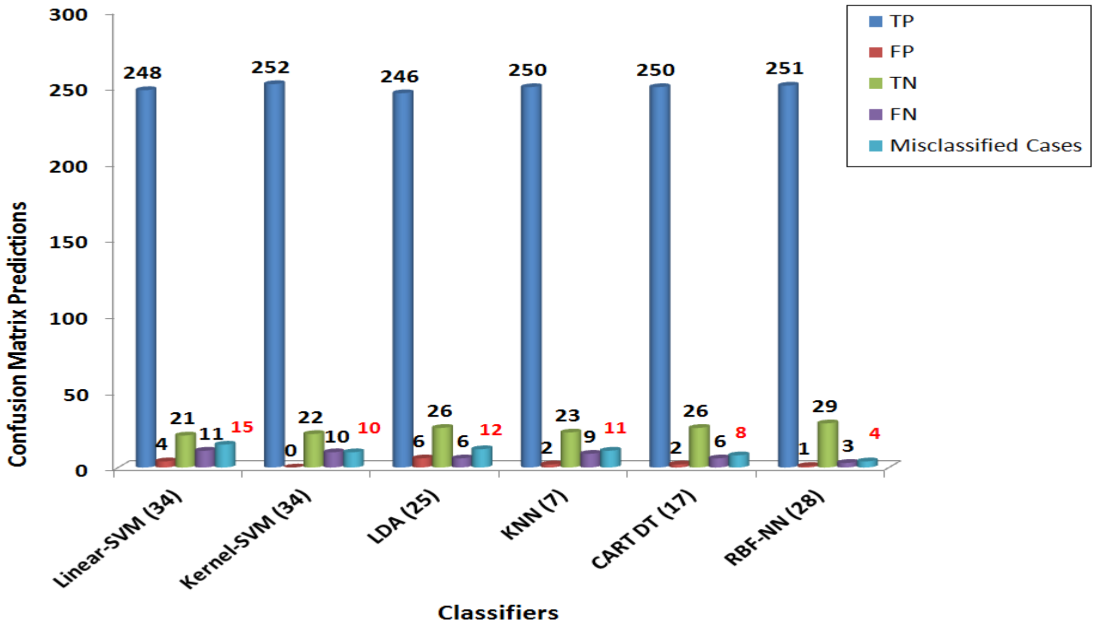


Figure 5.8: Graphical illustration of confusion matrix and misclassified instances of different classification modules (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT, RBF-NN) using the best selected features as represented in brackets

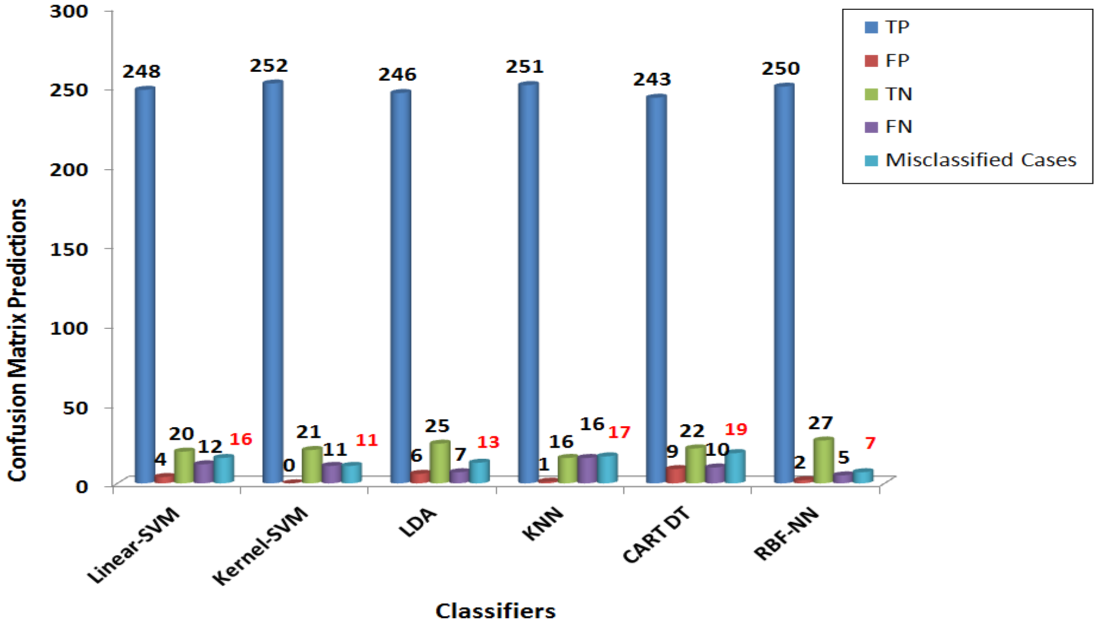


Figure 5.9: Graphical illustration of confusion matrix and misclassified instances of different classification modules (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT, RBF-NN) using all (40) features.

Figures 5.10 and 5.11 illustrate the results of the classification performance measures (Recall, Specificity, B-Accuracy, Precision, F-score and MCC) using the best selected feature sets and all feature sets, respectively. The followings analyse and discuss these performance measures individually.

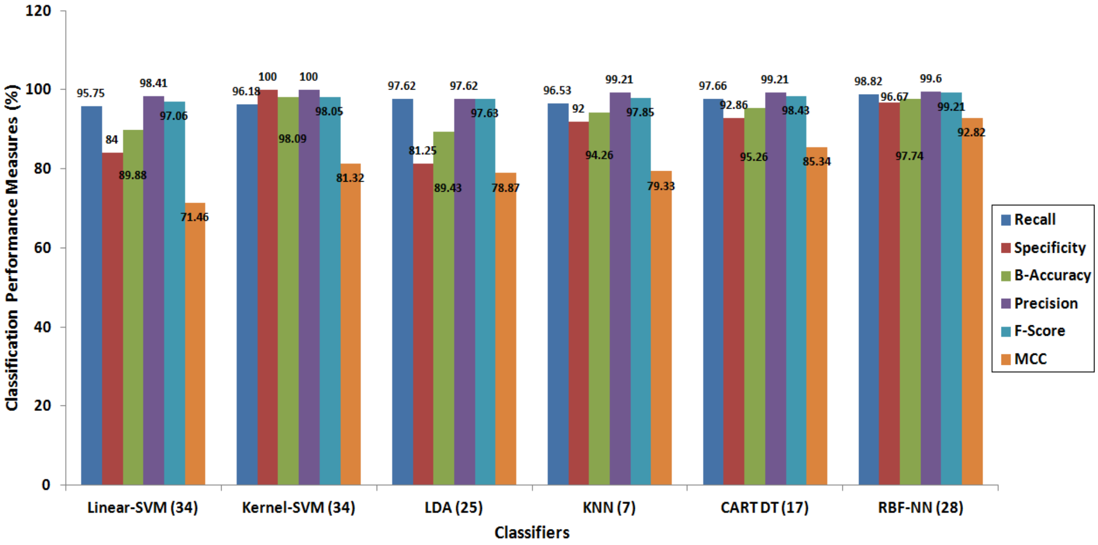


Figure 5.10: Classification performance measures using the best selected feature sets

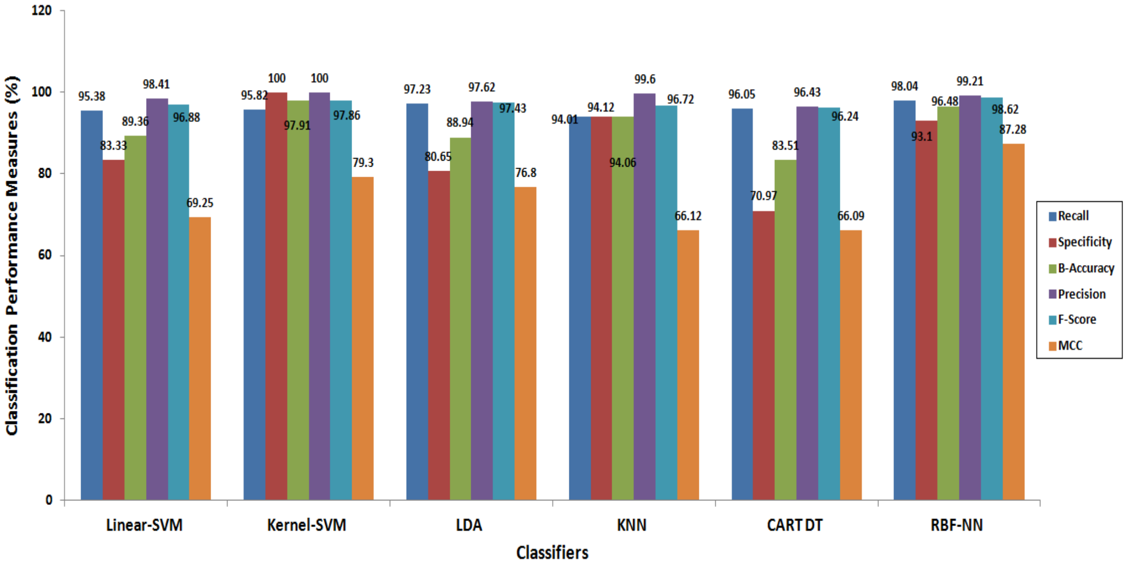


Figure 5.11: Classification performance measures using all feature sets

Regarding recall results, it is clearly evident from Table 5.5 and Fig. 5.10 that the best recall value for the top selected feature sets was achieved by the RBF-NN classifier, with 98.82% followed by the CART DT with 97.66% and the LDA classifier with a value of 97.62%. This is due to the low number of abnormal PF subjects that have been misclassified as normal PF class (FN=3). The lowest recall measure was achieved by Linear-SVM, this can be interpreted by the large number of false negative predictions (FN=11). For all features (Table 5.5 and Fig. 5.11), the best recall value was also achieved by RBF-NN with 98.04% followed by LDA and CART DT classifiers with a score of 97.23% and 96.05%, respectively. Additionally, as shown in Table 5.5, Figures 5.10, and 5.11, the perfect specificity value with the best selected features was achieved by the Kernel-SVM classifier, with 100% (with zero FP misclassified predictions) followed by the RBF-NN model with 96.67% and the CART DT classifier with 92.86%. The lowest specificity measure was achieved by LDA classifier with 81.25%. This means a large number of false positive predictions (FP=6). For all features, the perfect specificity measure was also assigned to the Kernel-SVM classifier followed by KNN and RBF-NN with values of 94.12% and 93.10%, respectively.

For the best selected features, the highest B-Accuracy value was achieved by Kernel-SVM classifier with 98.09%, followed by RBF-NN with 97.74% and CART DT with 95.26%. While the lowest B-Accuracy value was achieved by LDA with 89.41% followed by Linear-SVM classifier with 89.88%. For all features, the best B-Accuracy measure was achieved by Kernel-SVM with 97.91%, followed by RBF-NN and KNN with a measure of 96.48% and 94.06%, respectively.

In terms of high precision score using the best selected features, the kernel-SVM was the perfect model with 100% score, followed by RBF-NN with 99.61%, CART DT and KNN models with a score of 99.21%. The variation in precision between the kernel-SVM model and other models can be interpreted by the high number of false positives predicted by other models. For all features, the kernel-SVM was also the perfect model with 100%, followed by KNN with 99.60%, RBF-NN with a score of 99.21%.

The F-score results using the best selected features suggested that the RBF-NN classifier was the best with 99.21%, followed by the CART DT with 98.43% and the Kernel-SVM classifier with a score of 98.05%. For all features, the RBF-NN achieved the highest F-Score with 98.62%, followed by the CART DT and Kernel-SVM with a score of 98.43% and 98.05%, respectively.

By examining the previous result using the best selected features, the highest MCC value was achieved by the RBF-NN classifier with 92.82%, followed by the CART DT with 85.34% and the Kernel-SVM with a value of 81.32%. The worse MCC value was achieved by linear-SVM with 71.46% followed by LDA model with a value of 78.87%. For all features, the RBF-NN was also the best model with a value of 87.97%, followed by the Kernel-SVM and CART DT with a value of 79.80% and 76.80%, respectively. From Table 5.5 and Fig. 5.12 using the best selected features, the best average measure was achieved by the RBF-NN classifier with 97.48%, followed by the Kernel-SVM with 95.61% and CART DT with 93.20%. For all features, the RBF-NN was also the best model with a value of 95.46%, followed by the Kernel-SVM with 95.15% and KNN with 90.77%.

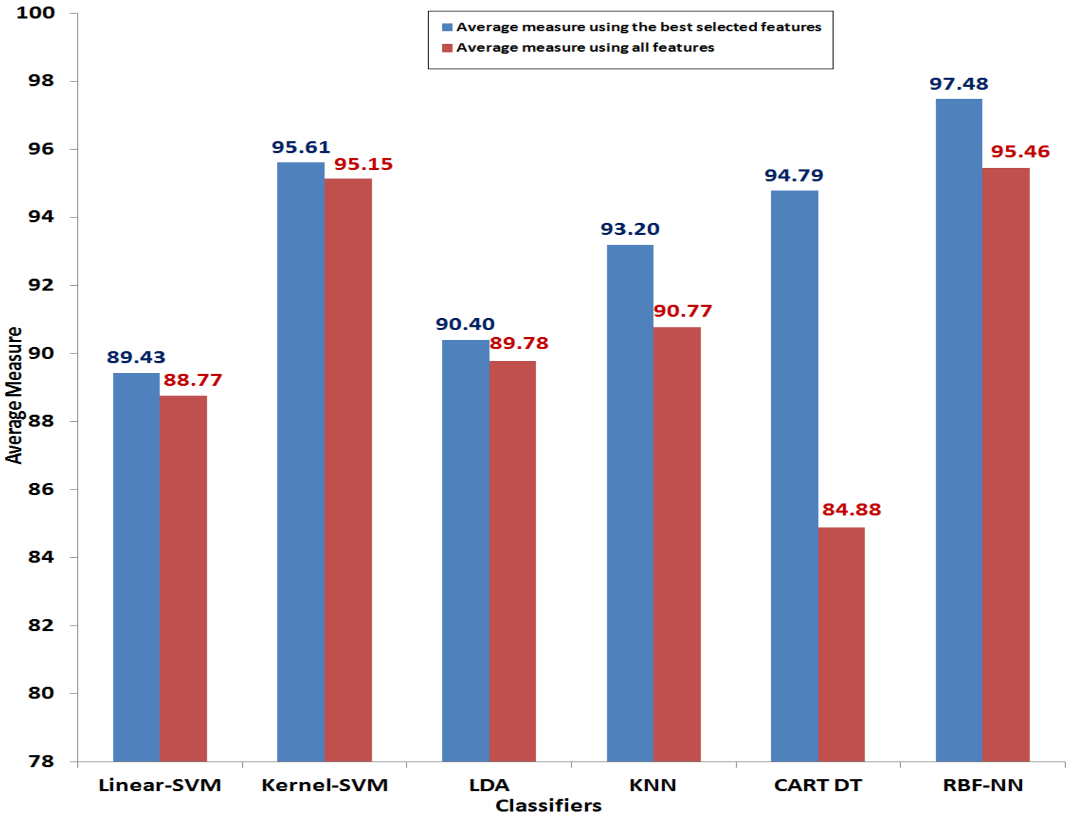


Figure 5.12: Average performance measure for different classifiers using different selected features (showing differences between classification models using the best selected features and others using all (40) features).

Figure 5.13 illustrates the AUC result for different classifiers using different selected features. This clearly indicates the differences between classification models using the best selected feature set approach and others without it.

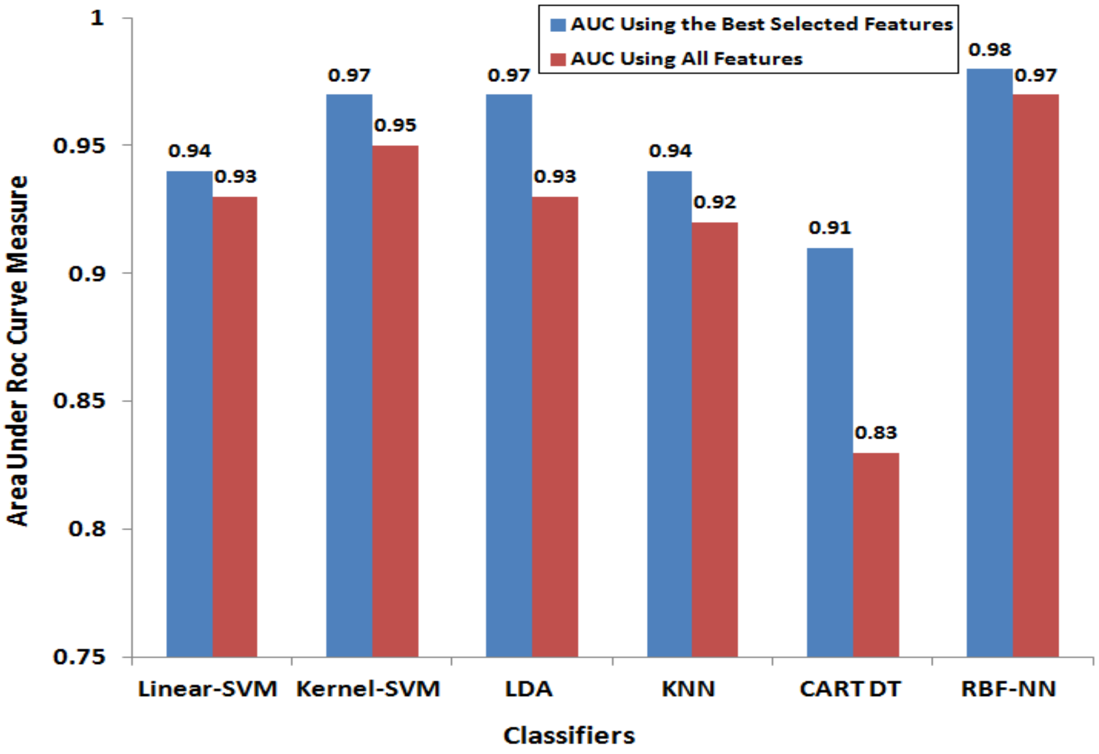
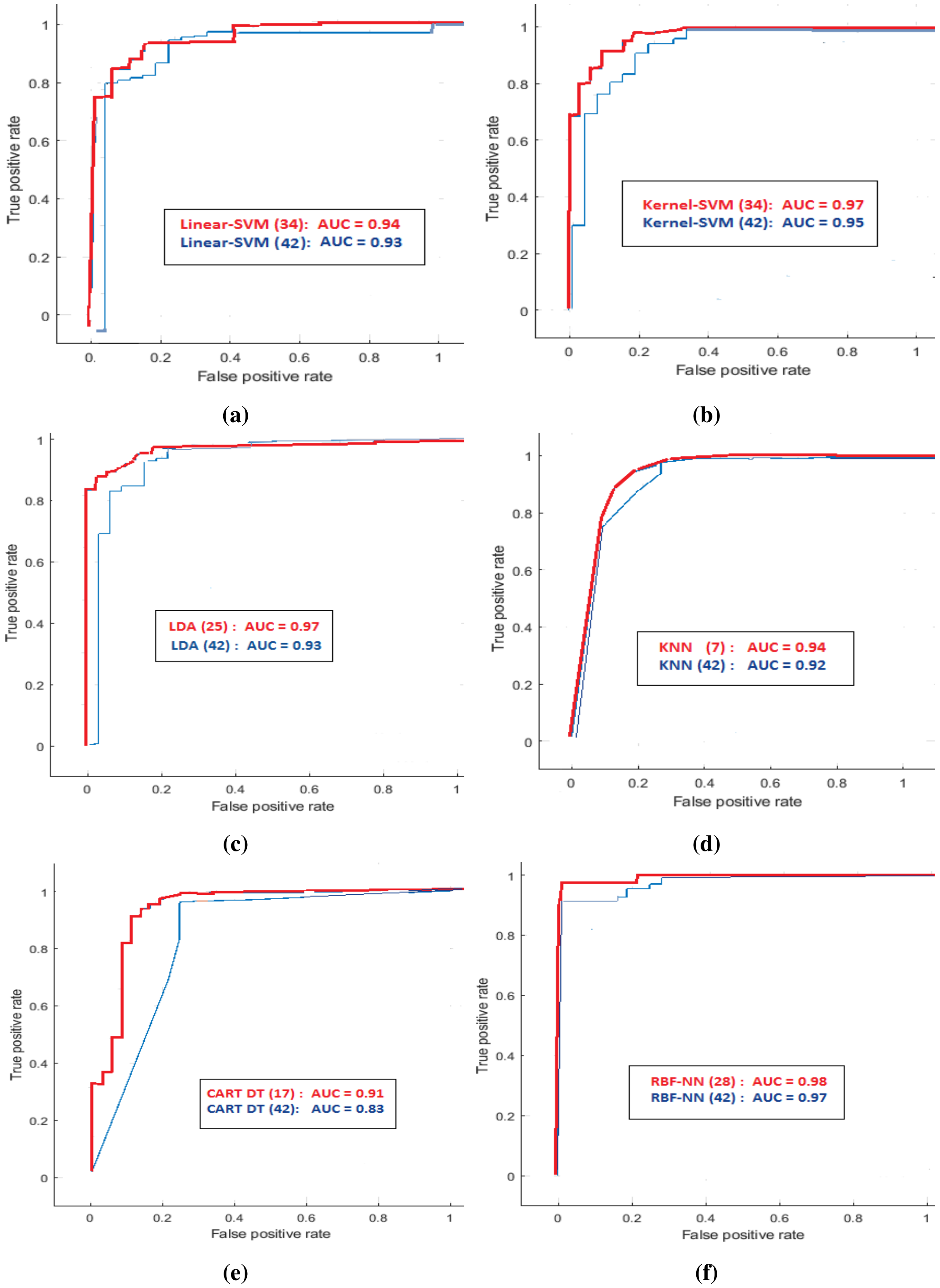


Figure 5.13: Area under ROC curve classification performance measure for different classifiers using different selected features (showing differences between classification models using the best selected features and others using all (40) features).

Fig. 5.14 shows the ROC curves analysis for all classification modules using the best selected features (red) and all (40) features (blue). From Figures 5.14 and 5.13, the AUC was the best for RBF-NN classifier with 0.98, followed by Kernel-SVM and LDA classifiers with a score of 0.97 when using the best selected features. For all features, the highest AUC was achieved by RBF-NN with 0.97 followed by Kernel-SVM, Linear-SVM and LDA with a score of 0.95, 0.93 and 0.93, respectively.



(a) (b) (c) (d) (e) (f)

Figure 5.14: ROC curves analysis showing AUC measures for different classifiers using the best selected features (red) and all selected features (blue). The number inside brackets indicate the number of selected features. (a) the Linera-SVM classifier, (b) the Kernel-SVM (c) the LDA, (d) the KNN, (e) the CART DT and (f) the RBF-NN classifier.

When considering the average classification execution time for different classifiers (Table 5.5 and Figure 5.15) using the best selected feature sets, the RBF-NN model took up the least execution cost with a negligible time of (0.18 sec) followed by CART DT with (0.3 sec) and LDA with (0.31 sec). For the classification execution cost using all extracted features the RBF-NN model was also the best with (0.21 sec) followed by KNN with (0.37 sec) and CART DT with (0.4 sec).

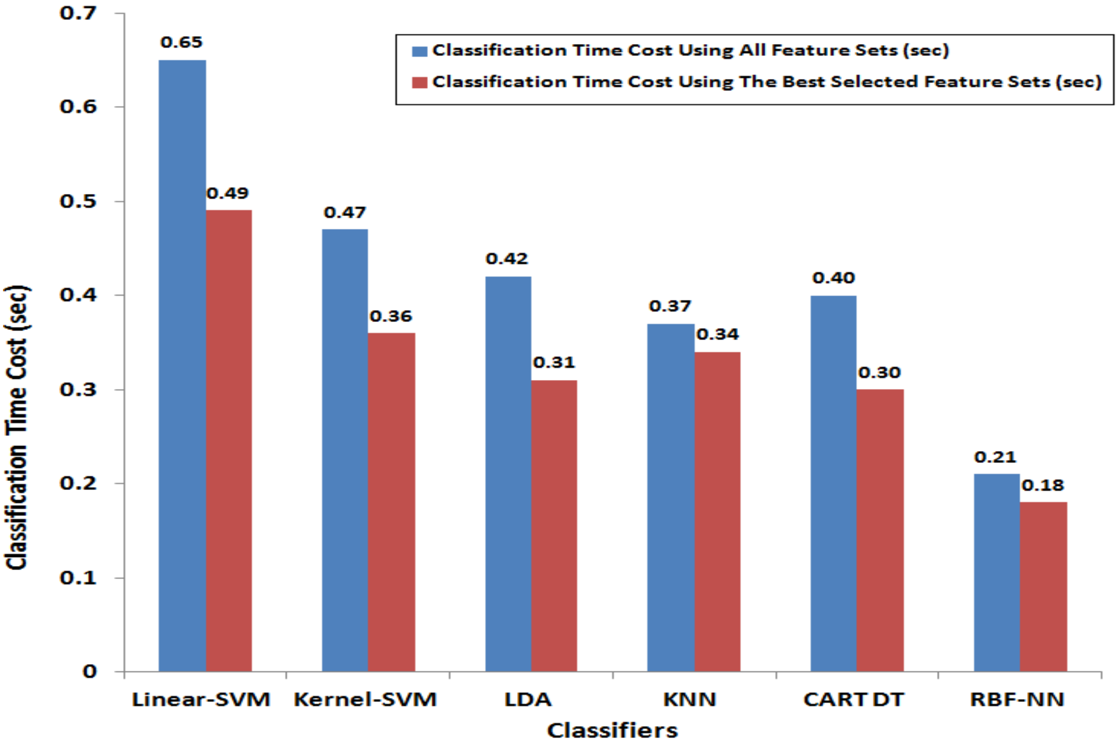


Figure 5.15: Classification time cost for different classifiers using the best selected features and all features

It is clearly evident from the above results (Table 5.5) and the visual representation (5.8, 5.9, 5.10, 5.11, 5.14, 5.13, and 5.15) that the performance of most of classification models has been improved when using the best selected features. This also indicates the usefulness of using feature selection methods for improving the classification performance and the execution cost. Below, we present our discussion in selecting an appropriate classification model for the supervised ultrasound PF classification approach.

A common question in most classification tasks is how to select the best model for a specific problem. The easy way is to calculate the accuracy of different classifiers on a specific dataset and choose the most accurate one. However, the accuracy of a classifier alone is typically not enough for selecting the best model. This is why in this study, we introduced 7 different evaluation metrics based on the type of dataset (imbalanced dataset); three weak measures (Recall, Specificity and Precision) and four strong measures (B-Accuracy, F-Score, MCC and AUC). The weak measures used in the classification evaluation were unable to capture the whole confusion matrix class attributes even when giving a perfect measure score. This is due to the short number of confusion matrix attributes used in the measures computation (only two attributes: TP and FN for Recall, TN and FP for Specificity, and TP and FP for Precision). This also makes the performance evaluation inaccurate when dealing with imbalanced datasets as this was the case in this study. Therefore, introducing only these measures in the evaluation is typically not enough to decide which model performs better. Thus, the remaining measures (B-Accuracy, F-Score, MCC and AUC) would be more appropriate for the evaluation. In terms of high B-Accuracy measure, the Kernel-SVM was the best model for all different extracted feature sets, followed by the RBF-NN model. For F-score, MCC and AUC measures using the best selected feature set, the RBF-NN model achieved the highest results followed by the CART DT and the Kernel-SVM models. But when using all extracted feature sets, the Kernel-SVM model performs better than the CART DT model. This indicates that the Kernel-SVM classifier performs better when using larger feature set, while the CART DT model achieved its best with a small selected feature set. In general, both RBF-NN and Kernel-SVM outperform other models in terms of high B-Accuracy score and high AUC score when dealing with all different selected feature sets. However, it is clearly evident from the overall results presented in Table 5.5, that the RBF-NN is superior than the Kernel-SVM model and achieved its best in terms of four high performance measures (Recall, and F-Score, MCC and area under ROC curve) and low classification execution time for all different selected feature sets. In order to derive a final decision in defining the best classification model, an average measure of all 5 performance measures including Recall, Specificity, B-Accuracy, Precision, F-score and MCC was computed along with the consideration of AUC score and the execution time cost as presented in Table 5.5 and Fig. 5.12. Finally, it is concluded from the above results that RBF-NN classification model achieved the best results in differentiating between asymptomatic and symptomatic PF subjects in terms of: (i) low misclassified instances (4 out of 284 for the best selected features and 7 out of 284 for all features); (ii) high performance measures for Recall, F-Score, MCC and area under ROC curve; (iii) best average measure with 97.48% and 95.46% using the best selected features and all features, respectively; (iv) and low classification execution time for all different selected feature sets. These best results were achieved by the RBF-NN using both the best 28 selected features and all 40 features (represented in Table 5.4 with only 20 nodes in the hidden layer. This also demonstrates the flexibility (in choosing the parameters), the good implementation of the RBF-NN classifier, which can often lead to good results and reproducibility in terms of high performance when compared to other classification models. Thus, the RBF-NN is recommended in the classification of normal and abnormal ultrasound images of the plantar fascia region. Furthermore, some previous studies had successfully introduced feature selection methods and RBF-NN classification module in classifying different ultrasound images (Horng, 2009, 2010, 2013) with satisfactory results. Similarly, this study has proven the use of feature selection approach (for features characterization and and selection) and RBF-NN classification module to improve the performance of the classification predictions and the execution time, and to differentiate between asymptomatic and symptomatic PF subjects in identifying the patients that are at risk of plantar fasciitis.

5.6 Summary

In this study we developed a new automatic supervised classification system for discriminating different ultrasound plantar fascia images. Six different feature set measures were used to extract and analyse the texture features. Additionally, the infinity selection method was successfully adopted to rank and characterize asymptomatic and symptomatic features, based on their weights importance. The results of the feature selection stage revealed that the top 28 selected features can represent the characteristics of asymptomatic and symptomatic PF subjects ultrasound images well. The Inf selection method to select the best features is quite effective. In order to define and compare the best features, the F-score measure was independently computed for different classifiers, Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN using different selected feature sets (1-40). The best selected feature set for every classifier were fed to the related classifier as the input vector for the classification task. In the experiments, different performance evaluation measures were used to assess the classification capability of the six classifiers using their best selected features. This includes, confusion matrix, Recall, Specificity, B-Accuracy, Precision, F-score, MCC, ROC curve and AUC, and execution time cost. The experimental results demonstrated that the RBF-NN models was superior than the other five models (using only 28 top selected features). This also indicates the effectiveness of the RBF-NN model when introduced in the classification and the discrimination of different ultrasound plantar fascia images. Even though the RBF-NN model can effectively introduced in the classification and the discrimination of different ultrasound plantar fascia images, further investigation is required for future research, such as, the integration of other texture feature measures and other selection methods using deep learning approach and 3-D representation, to provide a better representation of different PF structures (rearfoot, midfoot and forefoot) and to improve the characterization and the discrimination of different PF subjects. Furthermore, the above results should be verified with more patients and more clinical datasets, asymptomatic and different symptomatic classes, plantar Fasciitis, plantar fascial tears, plantar fibromatosis. In conclusion, the results in this study showed that it is possible to discriminate a group of patients with plantar fasciitis based on texture features (extracted from US images of plantar fascia) and feature selection analysis using different classification modules.

Chapter 6

Novelty detection model for ultrasound images of the plantar fascia tissue

6.1 Overview

Due to the lacking in the dataset to describe PF US abnormality behaviour (i.e. dataset imbalanced) in this study, this chapter introduces an automatic novelty detection model (one-class classification model) applied to the PF ultrasound images dataset (using 252 normal PF subjects) for detecting plantar fascia abnormalities embedded in normal datasets and draw an abnormality threshold that separate the normal samples from abnormal ones. Different novelty models have been investigated in the literature for novelty detection in medical datasets. The model uses support vector data description (SVDD) based approach to define plantar fascia abnormalities behaviour (i.e. estimating novelty scores or thresholds that separate normal and abnormal classes). These thresholds were computed using only the training normal datasets, testing normal data, and testing abnormal data. The optimal novelty threshold is defined by the testing and the validation of PF datasets. This will also help in detecting early warning of plantar fascia abnormalities such as plantar fasciitis. The proposed model introduces the following modules: (1) suitable feature extraction, selection and normalisation methods (defined earlier from chapter 5), (2) a classification novelty detection based technique (One-class SVDD classification module) applied to the normal plantar fascia ultrasound images dataset, and (3) the performance of the one-class novelty model is evaluated using B-Accuracy, F-score, MCC percent, ROC plot, AUC (defined earlier in Chapter 5), Gmean measure and time cost. The effectiveness of the model is evaluated in terms of high B-Accuracy, F-score, MCC , AUC and Gmean, and low time complexity.

6.2 Novelty Detection Background

6.2.1 Novelty Detection Description

The novelty detection technique is also known as anomaly detection, outlier detection technique (i.e detecting abnormal samples lying outside the majority of normal samples in the feature space) (Ritter and Gallegos, 1997) or one-class classification technique (i.e. only normal data is used during the training phase) (Moya and Hush, 1996). The novelty detection concept can be defined as a binary classification task that discriminates between test data samples (symptomatic dataset not used during the training stage) and the initial well sampled normal dataset (available during training). This concept is mainly applied to datasets in which a large number of normal samples exist and there is a lacking in datasets to describe abnormality behaviour (i.e. the dataset is imbalanced, as it was the case in this study). As the imbalanced dataset alters the performance outcomes of the most standard classification methods, presuming the feed in datasets are well distributed and balanced (Japkowicz and Stephen, 2002). This leads to the assumption that most novelty detection and one-class classification approaches are more appropriate for imbalanced datasets (Kennedy et al., 2009).

6.2.2 Novelty Detection Approach and Applications

Generally, the novelty detection approach tends to learn and describe the normality of a given dataset (assumed to be very well sampled) by building a model for representing the asymptomatic instances behaviour and detecting the novelty score (decision threshold). The new datasets (not very well sampled) are then tested and compared with the previously built model, and if the decision score is exceeded the tested dataset are then considered as symptomatic (irregular). Novelty and anomaly detection approaches have been widely introduced in many application fields such as medical diagnostic issues (Quinn and Williams, 2007), detecting masses in mammograms (Tarassenko et al., 1995), structural damage (Surace and Worden, 2010), text data mining (Basu et al., 2004), fault detection (King et al., 2002), and others.

6.2.3 Medical Novelty Detection Approaches

In this study we are only concerned with the medical image processing novelty detection domain. The medical imaging data can have some anomalies due to several causes, including, abnormal patient condition and characteristics, machine faults or data acquisition errors. Some of the novelty detection approaches dealing with the medical data can be categorized into three main groups, probabilistic-based, neighborhood-based (or distance-based) and classification-based (machine learning) novelty detection techniques (Campilho and Karray, 2016).

6.2.3.1 Probabilistic-Based Approach

The probabilistic-based approach (also called statistical-based approach) is statistically based on the probability of detecting novel (or abnormal) cases in a specific dataset using density estimation methods (PDF) to draw a thresholded separation space between normal and novel (or abnormal) samples such as low density regions in the training samples shows high chance of containing novel (or abnormal) instances (Silverman, 2018; Pimentel et al., 2014). As an example, parametric gaussian mixture model (GMM) (McLachlan and Basford, 1988), nonparametric kernel parzen windows estimator (PARZEN) (Parzen, 1962), and gaussian process one-class regression estimator (GPOC) (Kemmler et al., 2010) were very common methods to estimate the PDF for building the normality dataset models (Shental et al., 2003; Breaban and Luchian, 2012; Park et al., 2010), . However, the performance of these methods are affected by the small amount of training samples (Pimentel et al., 2014).

6.2.3.2 Neighbourhood-Based Approach

Unlike the probabilistic-based approach, Neighbourhood-based methods are unsupervised in nature (i.e. do not take into consideration any prior knowledge about the dataset space distribution) (Campilho and Karray, 2016). They were mainly based on the nearest-neighbour and clustering analysis concept, where the normal dataset samples are grouped together, while novel dataset (or anomalies) placed outside their nearest neighbours (Pimentel et al., 2014) using different novelty scores estimation approaches such as Euclidean distance measure (Tan et al., 2005) and k-means clustering method (Jain and Dubes, 1988; Tan et al., 2005) for grouping together normal dataset instances in different clusters while the abnormal (or novel) instances were placed outside these clusters (Chandola et al., 2009). However these methods are computationally expensive and their performance depend mainly on the use of distance-based and clustering-based novelty score computation methods (Chandola et al., 2009; Campilho and Karray, 2016; Pimentel et al., 2014).

6.2.3.3 Machine Learning Classification-based Approach

Machine learning classification based novelty detection techniques such as neural network and support vector machines are very common and widely introduced in the one-class classification approaches (Chandola et al., 2009; Campilho and Karray, 2016; Pimentel et al., 2014). For the neural network novelty detection approach, the neural network is trained using only normal dataset instances and tested with both normal and abnormal samples. The testing inputs are either classified as normal or novel data instances (De Stefano et al., 2000). Different neural networks techniques have been used in the literature for novelty detection (Chandola et al., 2009) including multi layered perceptrons (back-propagation neural network) (Augusteijn and Folkert, 2002), neural trees (Martinez, 1998), auto-associative networks (Aeyels, 1991), adaptive Resonance theory based (Moya et al., 1993), radial basis function (Bishop, 1994; Albrecht et al., 2000), hopfield networks (Jagota, 1991) and oscillatory networks (Ho and Rouat, 1997; Tuong Vinh and ROUAT, 2001), the Self-Organising Map (SOM) (Kohonen, 1982, 1990). On the other hand as reported by (Clifton et al., 2008), the Support Vector Machine (SVM) has been considered as a good successor to the neural network, whith the conception of finding an optimal hyperplane to separate dataset attributes into different classes after using a well defined kernel function to learn complex class areas and their boundaries (Vapnik, 2013). The One-class SVM approach (Schölkopf et al., 1999) has been widely introduced in various medical and non-medical data analysis for novelty and anomaly detection applications, such as seizures detection and analysis in humans' intra-cranial (Gardner et al., 2006), anomaly detection in normal time-series points (Ma and Perkins, 2003), novelty detection techniques in gas-turbine engines and industrial systems (Clifton et al., 2008). However, the One-class SVM approach is computationally complex due to the complexity of the kernel functions (Pimentel et al., 2014). As a solution to this, Tax and Duin (1999b) proposed a support vector data description (SVDD) approach that employs automated parameters optimisation using linear kernels instead of the quadratic kernels (Campbell and Bennett, 2001; Tax and Duin, 1999a). Furthermore, the proposed solution using SVDD model is more suitable for unbalanced datasets (as it was the case in this study) with just a small number of abnormal cases. Other proposed solution and approaches can be found in (Pimentel et al., 2014).

Therefore, the main contribution of this study is to introduce and demonstrate a suitable novelty detection based model to perform one-class classification for plantar fascia analysis in ultrasonic images, mainly based on novelty detection and one-class classification performance evaluations. So, the general objective of this study is to derive for the first time a new suitable novelty detection model for PF ultrasound images dataset, based primarily on data from asymptomatic samples and and SVDD classifier proposed in some previous studies (Pimentel et al., 2014) aiming to perform applicable classification efficiencies in plantar fascia real world data acquired using ultrasound systems. The proposed model uses the one-class classification approach of previously defined asymptomatic samples, in combination with the detection of novel (symptomatic) plantar fascia samples. The developed system applies the followings: preprocessing, PF segmentation, features extraction, ranking and discriminative features selection, features analysis, PF US images classification using SVDD classifier. The effectiveness of the SVDD novelty model is evaluated and compared to GMM, PARZEN, GPOC, and Self-Organising Map (SOM), using six novelty detection performance measures including B-Accuracy, F-Score, MCC, Gmean, AUC and Time cost.

6.3 Novelty Detection Model

In this section, we describe and discuss the proposed novelty detection model for detecting the symptomatic PF samples (with plantar fascia disorder) under the designed novelty detection framework.

The proposed PF Novelty detection model implies the following steps as illustrated in Figure 6.1: (1) preprocessing phase applying normalization and feature selection (using an unsupervised infinity feature selection method (Roffo et al., 2015b)) to the initial features extracted previously from the PF segments (28 features). The aim of this phase is to normalize, select and analyse the most discriminating and suitable features for the novelty detection model. It scales all the features in the same range [-1 1], so that all features can be treated as equally important a priori and gets rid of all redundant features and preserves information about normal PF instances. (2) Building a normality model by training novelty detection classifier (SVDD) using only asymptomatic (normal) PF training datasets. (3) Estimating novelty scores using the trained ND classifiers, training and testing asymptomatic datasets, and testing symptomatic datasets of the PF dataset. (4) classification performance analysis to compare the model with other approaches

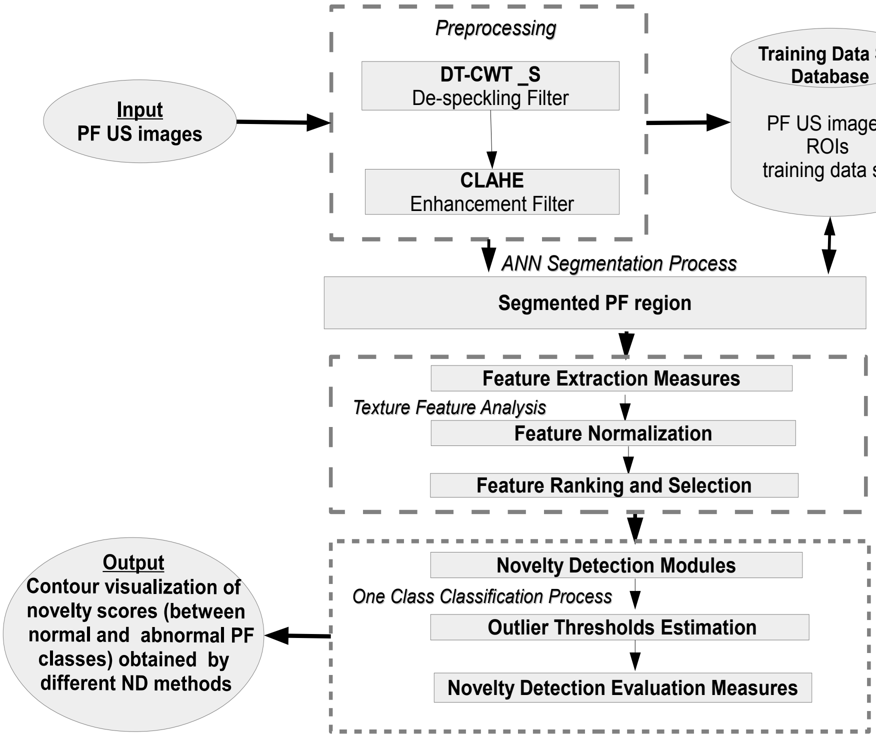

Preprocessing
DT-CWT _S
De-speckling Filter
CLAHE
Enhancement Filter
Input
PF US images
Training Data Set
Database
PF US images
ROIs
training data set
ANN Segmentation Process
Segmented PF region
Feature Extraction Measures
Texture Feature Analysis
Feature Normalization
Feature Ranking and Selection
Novelty Detection Modules
One Class Classification Process
Outlier Thresholds Estimation
Novelty Detection Evaluation Measures
Output
Contour visualization of novelty scores (between normal and abnormal PF classes) obtained by different ND methods

Figure 6.1: Flowchart illustration of the plantar fascia novelty model based on a Texture features analysis and SVDD outlier detection module

introducing 6 (B-Accuracy, F-Score, MCC, Gmean, AUC and Time cost) different performance measures. complexity.

6.4 PF ultrasound data description

Various PF US images, acquired from 45 patients' footprint area (rearfoot, midfoot and forefoot sections) in the prone position were used in the novelty detection approach (Figure 6.2); more specifically, a total of 284 PF US images (252 normal and 32 abnormal taken from diabetic patients with plantar fasciitis). These images were initially preprocessed using DT-CWT_S and CLAHE filters to reduce the speckle noise effect and to enhance the PF region contrast, respectively, and they were automaticaly segemented using RB-ANNs supervised segmentation approach (Boussouar et al., 2017a) described earlier in Chapter 4 and 5.

All the methods used in the proposed approach were implemented using Matlab R2017b (The MathWorks Inc., Natwick, USA).

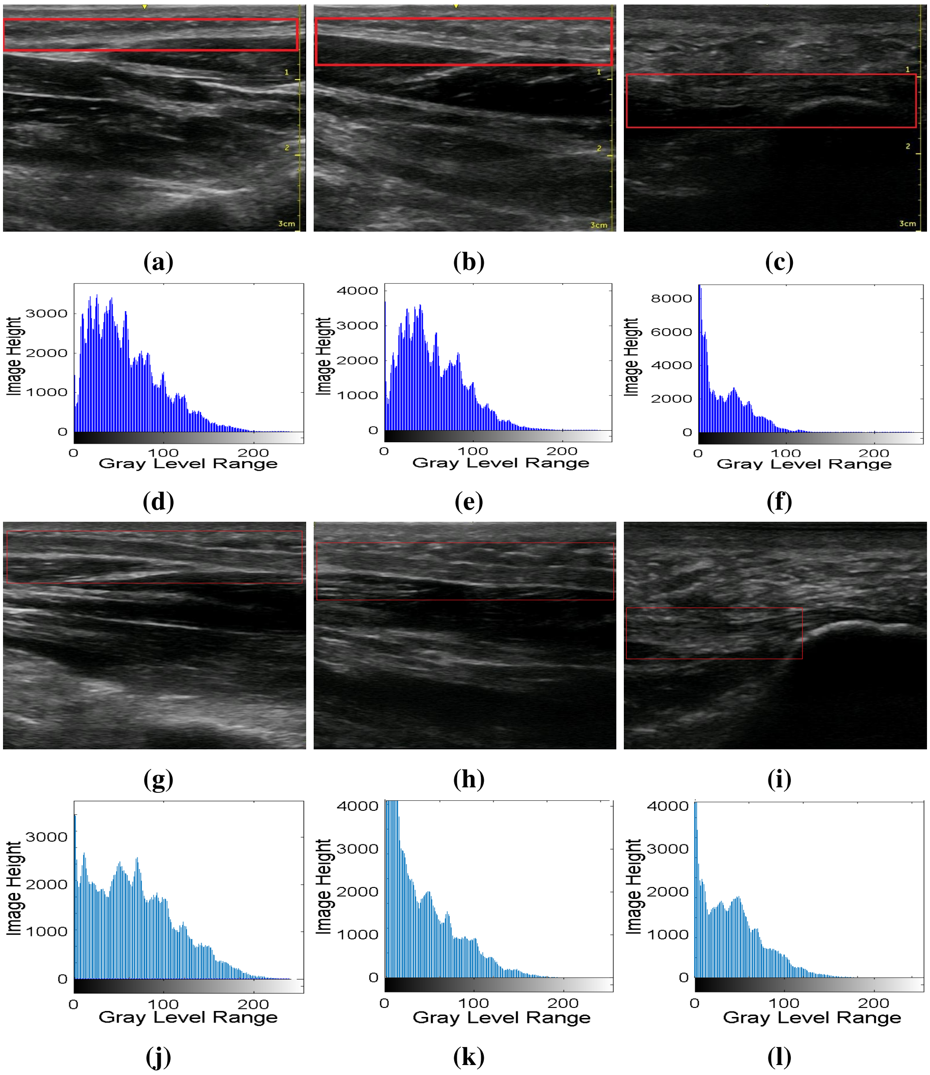


Figure 6.2: US images for different PF structures: (a)-(c) Asymptomatic PF US samples (Forefoot, Midfoot and Rearfoot sections, respectively); (d)-(f) their normal gray level histogram representation; (g)-(i) Symptomatic PF US samples (Forefoot, Midfoot and Rearfoot section, respectively); (j)-(l) their abnormal gray level histogram representation.

6.5 Feature analysis

Based on our previous study results described in Chapter 5, only 28 features extracted from the segmented PF region have been proven especially effective for the characterization of the PF region. This includes: (i) SGLDM features calculated and averaged for a selected distance $d = 1$ (3×3 matrices) and four various orientation angles $\theta = 0^o, 45^o, 90^o$, and 135^o, (ii) region based features, (iii) NGTDM features for a 3x3 kernel size, (iv) FOS features (Umbaugh, 2005; Christodoulou et al., 2003), (v) SFM features (Wu and Chen, 1992), and (vi) Laws' texture energy measures TEM. All the calculated feature measures are presented Table . More details about these features can be found in Chapter 5. In order to overcome the problem of the high dimensionality of the extracted features (28 features × 284 samples) and to select the most discriminating features for novelty detection model, features ranking and features selection approaches were also introduced using unsupervised infinity technique proposed by Roffo (2015a; 2017). This will also minimize the computation cost and prevent over-fitting problems. The PF ultrasound datasets consist of both asymptomatic and symptomatic samples. However, only the asymptomatic samples were used for MVN normalisation (discussed earlier in Chapter 5) in order to build our normality model (where only asymptomatic data are needed to build PF normality model). On the other hand, the PF symptomatic samples were only used to test and validate our novelty detection model of normality.

Extracted Features

(i) SGLDM	(ii) Region Based	(iv) FOS	(vi) TEM
(1) angular 2nd moment, (2) contrast, (3) correlation, (4) sum of squares, (5) variance, (6) sum average, (7) sum entropy, (8) entropy, (9) difference variance, (10) difference entropy.	(1) major axis length (2) orientation (iii) NGTDM (using 3x3 kernel size) (1) contrast, (2) busyness, (3) complexity, (4) strength	(1) mean, (2) variance, (3) skewness, (4) kurtosis, and (5) energy (v) SFM (1) contrast, (2) periodicity, (3) roughness	(1) LL, (2) EE, (3) SS, (4) LE, (5) ES; (6) LS

Table 6.1: Extracted feature measures

6.6 SVDD novelty detection algorithms

In this study, SVDD novelty detection approach (Tax and Duin, 1999b; Pimentel et al., 2014) has been selected to minimize the hypersphere margin. The hypersphere of SVDD approach is defined by its center **c** and radius **R** surrounding the training dataset labelled as $T_i, i = 1, ..., N$, where N represents the total number of training samples. Generally, the structural risk of SVDD can be formulated using equation (6.1) and minimized by equation (6.2) when all the training instances are distributed only within one hypersphere.

$$Y(R,c) = R^2. \tag{6.1}$$

$$||x_i - c||^2 \leq R^2, \forall i. \tag{6.2}$$

On the other hand, when the outliers (novelties) exist in the training datasets, the minimization formulation is represented by (6.3) and minimized by (6.4) (minimizing R, ξ and constant C_T).

$$Y(R,c,\xi) = R^2 + C_T \sum_{i}^{N} \xi \tag{6.3}$$

$$\|x_i - c\|^2 \leq R^2 + \xi_i, \xi_i \leq 0, \forall i, \tag{6.4}$$

where R and c represent the radius and the centre of the sphere, respectively, ξ denotes slack variables (introduced here to permit some training dataset instances outside the hypersphere). C_T is a constant monitoring the trade-off between the volume of the hypersphere and the one-class classification rejected-errors. In order to solve the previous optimization the Lagrangian dual problem (L_{DP}) is introduced here to maximize equation (6.5) subject to the constraint presented in equations (6.6) and (6.7).

$$L_{DP} = \sum_{i}^{N} \alpha_i (x_i.x_i) - \sum_{i,j}^{N} \alpha_i \alpha_j (x_i.x_j) \tag{6.5}$$

$$0 \leq \alpha_i \leq C_T, \forall i. \tag{6.6}$$

$$\sum_{i}^{N} \alpha_i = 1. \tag{6.7}$$

Equation (6.5) can be replaced by equation (6.8), where $\Phi(x_i)$ performs the new feature space mapping.

$$L_{DP} = \sum_{i}^{N} \alpha_i \Phi(x_i).\Phi(x_i) - \sum_{i,j}^{N} \alpha_i \alpha_j \Phi(x_i).\Phi(x_j) \tag{6.8}$$

By using the Mercer kernel formula introduced by Vapnik (1998) where $\Phi(x_i)\Phi(x_j)$ is replaced by the kernel $K(x_i,x_j)$ (to exclude the mapping computation), equation (6.8) can be rewritten as follows:

$$L_{DP} = \sum_{i}^{N} \alpha_i K(x_i,x_i) - \sum_{i,j}^{N} \alpha_i \alpha_j K(x_i,x_j) \tag{6.9}$$

and the Lagrangian is calculated by (6.10) introducing the Gaussian kernel function as follows:

$$L_{DP} = \sum_{i}^{N} \alpha_i exp\left(\frac{-\|x_i - x_i\|^2}{\sigma^2}\right) - \sum_{i,j}^{N} \alpha_i \alpha_j exp\left(\frac{-\|x_i - x_j\|^2}{\sigma^2}\right). \tag{6.10}$$

where σ denotes the kernel width. In the previous new formulation a new instance z_i is distinguished as a novelty points (abnormal) if the following constraint is true and $z_i(x)$ is greater than the radius R:

$$\sum_{i}^{N} \alpha_i exp\left(\frac{-\|z_i - x_i\|^2}{\sigma^2}\right) < \frac{1}{2}\left[1 + \sum_{i,j}^{N} \alpha_i \alpha_j exp\left(\frac{-\|x_i - x_j\|^2}{\sigma^2}\right) - R^2\right]. \tag{6.11}$$

The Gaussian kernel function has been selected in this study, due to its strength of controlling the increased distances for multi-dimensional feature spaces (Tax and Duin, 1999a,b).

For SVDD novelty detection model threshold estimation and parameters setting, the regularization parameter C, and the width of the Gaussian kernel(σ) have to be set properly for the SVDD model. From equations (6.6) and (6.7), the upper and the lower limits for the user defined parameter C can be represented by $\frac{1}{N} < C < 1$ and C can be defined as

$$C = \frac{1}{NP} \tag{6.12}$$

where N is the number of training samples (Asymptomatic samples), P denotes the percentage of normal instances. In this work, the threshold of SVDD novelty detection model is set to the radius of the training data hypersphere R (i.e a new instance z_i is classified as abnormal if $z_i(x)$ are greater than the radius R, where $z_i(x)$ and R are computed from (6.12). The SVDD novelty scores (distances between novel points and the radius of the positive instances) are computed using the Euclidean distance-based method in the transformed feature space. In this work, P was set empirically to 0.1 and 0.5, the width of the Gaussian kernel σ was set to 0.5 and 0.34, and C was set to 0.06. The parameter σ controls the width of the Gaussian kernel, and hence the complexity of the SVDD. A threshold on the outputs of the SVDD is set to be the average output of the normal training data.

6.7 Novelty detection performance analysis

The performance and the effectiveness of the proposed novelty detection approach in detecting symptomatic novel points, were evaluated using the confusion matrix prediction instances described earlier in Chapter 5 to predict the following measures: B-Accuracy, F-Score, Matthew's correlation coefficient (MCC) (defined earlier in Chapter 5) and Gmean (Sokolova and Lapalme, 2009; Matthews, 1975; Compton and Cao, 2006; Liu et al., 2018). The performance of the model is evaluated in terms of high B-Accuracy, F-score, MCC, AUC and Gmean, and low time complexity. In this study the geometric mean (Gmean) (Kubat et al., 1998) computes the balance performance of a learning novelty detection module between the true positive rate (TPR) and the true negative rate (TNR). Gmean measure can be defined as follows:

$$Gmean = \sqrt{TPR \times TNR} \tag{6.13}$$

6.8 Novelty detection results and discussion

For the novelty detection experimental results, a total of 284 (252 symptomatic and 32 asymptomatic) US images of the PF regions were used. Based on the previous study presented in Chapter 5 only 28 best features were extracted both from symptomatic and asymptomatic US images of the PF segments. Feature normalization and selection approaches described in Chapter 5 were also introduced in this study and features means, weights and ranking orders were calculated and analysed for symptomatic and asymptomatic PF US images.

Figure 6.3 shows the distribution of the top two selected features (LS v. Contrast) of 284 PF dataset (252 asymptomatic in blue and 32 symptomatic in red) before and after scaling (normalization).

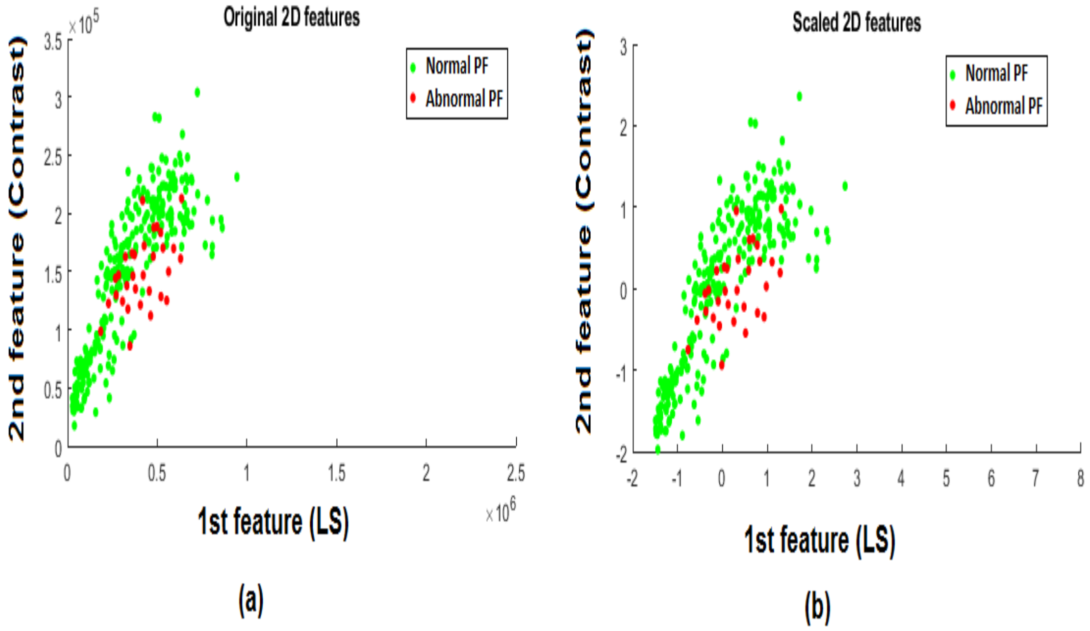


Figure 6.3: A 2-D plot representation of the top two selected features of 284 PF dataset (252 Normal and 32 Abnormal). Features of normal and abnormal PF samples are shown in blue and red, respectively. (a) PF features before scaling; (b) PF features after scaling (normalization). The plotting shows that the two selected features of the normal data and abnormal data lie in different range after scaling.

6.8.1 Feature selection analysis

Feature selection analysis results of the top 28 ranked features calculated from 284 US images of the segmented PF region are shown in Table 6.2. For each feature, the weight predictor was also computed and its rank order was assigned accordingly.

Table 6.2 summarises the feature selection results of the top 28 ranked predictors based on their weights importance for all dataset and their rank orders were assigned. The best features were found to be LS, Contrast, Variance, LE, Energy, SumSquare, AngSecMoment, LL, EE, DiffVariance, Strength, ES, Complexity, Correlation, DiffEntropy, SS, SumAverage, MajAxLength, Periodicity, Business, Mean, Skewness, Kurtosis, Orientation, Roughness.

In order to define the best selected features for each one-class classifier, the average Gmean measure was computed for different novelty detection modules (Parzen, GMM, GPOC, SOM and SVDD) using 28 different selected feature sets (starting with the highest ranked feature sets) as shown in Table 6.3. Five selected feature sets were defined using the best Gmean values. The best one-class classification Gmean results were achieved by SVDD using the the top 16 (highly ranked) features with a value of 0.873, followed by SOM using 15 features with a value of 0.843, the GMM using 7 features with 0.843, the GPOC using only 4 features with 0.81, and the PARZEN using 13 features with a Gmean value of 0.81.

Table 6.2: Feature selection analysis results of the top 28 selected features

Sellected Feature Sets	Predictor Important Weights	Predictor Rank Order
Haralick Spatial Gray Level Dependence Matrices (SGLDM)		
AngSecMoment	9.82	7
Contrast	7.78	20
Correlation	9.01	14
SumSquare	10.09	6
Variance	11.04	3
SumAverage	7.80	19
DiffVariance	9.38	10
DiffEntropy	8.73	17
Neighbourhood Gray Tone Difference Matrix (NGTDM)		
MajAxLength	7.55	21
Orientation	3.45	27
Histogram Features		
Contrast	11.22	2
Busyness	7.23	23
Complexity	9.27	13
Strength	9.38	11
Statistical Feature Matrix (SFM)		
Mean	5.62	24
Variance	8.88	16
Skewness	5.47	25
Kurtosis	3.62	26
Energy	10.53	5
Statistical Feature Matrix (SFM)		
Contrast	8.99	15
Periodicity	7.51	22
Roughness	3.11	28
Laws Texture Energy Measures (TEM)		
LL	9.77	8
EE	9.43	9
SS	8.71	18
LE	10.68	4
ES	9.29	12
LS	11.43	1

Table 6.3: Gmeans measure for different one-class classification modules using the top selected feature sets (1-28 sets) based on their weights and ranking orders.

Top Selected Features	**Gmean Measure for Different One-Class Classifiers**				
Selected Feature Set	**PARZEN**	**GMM**	**GPOC**	**SOM**	**SVDD**
1	0.500	0.242	0.415	0.350	0.250
2	0.676	0.637	0.619	0.685	0.685
3	0.504	0.718	0.702	0.810	0.548
4	0.685	0.552	**0.810**	0.765	0.747
5	0.676	0.740	0.776	0.740	0.776
6	0.559	0.468	0.702	0.573	0.726
7	0.468	**0.843**	0.713	0.718	0.843
8	0.342	0.810	0.747	0.776	0.776
9	0.622	0.468	0.625	0.839	0.656
10	0.622	0.530	0.747	0.726	0.843
11	0.619	0.419	0.612	0.839	0.803
12	0.573	0.573	0.656	0.637	0.781
13	**0.810**	0.433	0.718	0.702	0.637
14	0.523	0.433	0.740	0.776	0.781
15	0.530	0.354	0.593	**0.843**	0.685
16	0.781	0.500	0.390	0.776	**0.873**
17	0.718	0.500	0.750	0.747	0.781
18	0.713	0.354	0.612	0.839	0.776
19	0.685	0.354	0.726	0.713	0.718
20	0.776	0.433	0.433	0.713	0.685
21	0.390	0.433	0.541	0.702	0.740
22	0.530	0.250	0.500	0.637	0.661
23	0.530	0.250	0.354	0.573	0.619
24	0.559	0.250	0.500	0.619	0.573
25	0.438	0.250	0.433	0.596	0.573
26	0.438	0.350	0.433	0.593	0.573
27	0.306	0.354	0.250	0.405	0.405
28	0.395	0.500	0.250	0.573	0.726

6.8.2 Novelty Detection Classification analysis

In order to show the strength of the proposed model using SVDD one-class classifier with the proposed parameters settings, SVDD was compared to other novelty detection methods such as PARZEN (Parzen, 1962), GMM (McLachlan and Basford, 1988), GPOC (Kemmler et al., 2010) and SOM (Kohonen, 1982, 1990) implemented using Matlab Netlab toolbox (Nabney, 2002) and Novelty detection toolbox (Pimentel et al., 2014). Table 6.4 summarizes the parameters settings of all novelty detection methods based on the empirical results.

Table 6.4: Parameter setting values for each novelty detection method

Novelty detection method	Parameter setting values
SVDD	P = 0.1, margin distance = 2.5, C = 0.05, using Gaussian kernel function $\sigma = 0.34$ and $\rho = 0.06$
PARZEN	minimum neighbours = 4, neighbours fraction = 1/10, adaptive parzen window = false
GMM	kmeans iterations = 5, Expectationmaximization (EM) algorithm iteration = 80, type of the covariance matrices = spherical
GPOC	Kernel hyperparameters = [-1;-1.5], with mean measure
SOM	Ordering phase: Initial learning rate = 0.9, Final learning rate = 0.05. Initial neighbourhood size = 8, Final neighbourhood size =1
	Convergence phase: Initial learning rate = 0.05, Final learning rate = 0.01, Initial neighbourhood size = 0, Final neighbourhood size = 0

All novelty detection classifiers have been trained and tested using the same training and testing datasets, respectively. For normal PF US subjects, the dataset was split into 3 different classes 60% for training, 20% for validation and 20% for testing. On the other hand the abnormal PF subjects were split randomly into two classes: 50% for validation and 50% for testing (only normal data were used for training). Six different one-class classification performance measures were calculated and analysed using a 10-fold cross validation approach. For each novelty detection module the experiment was iterated 10 times on the PF US dataset and the the mean value was computed. The confusion matrix was estimated using testing data and the performance results of each novelty detection method were presented in Table 6.5.

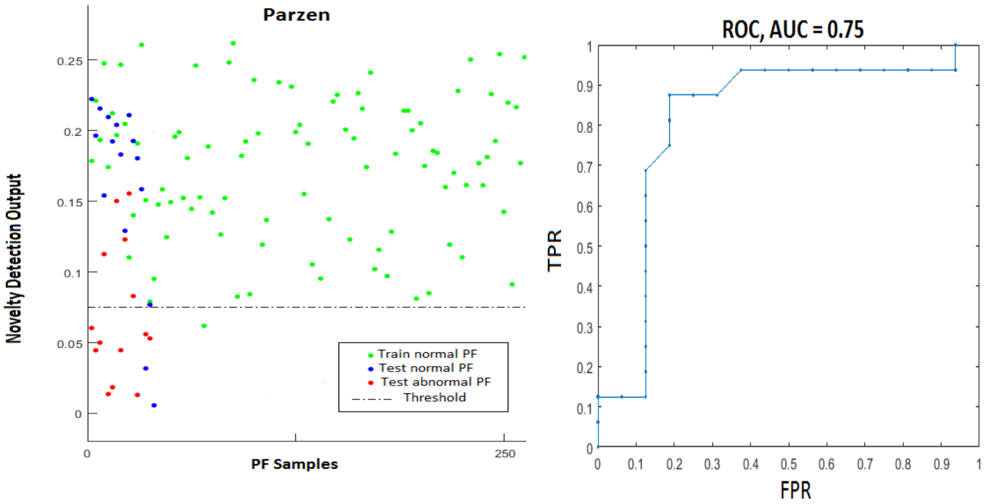
Table 6.5 tabulates the one-class classification performance results using different performance measures (B-Accuracy, F-score, MCC percent, Gmean, ROC plot AUC and Time Cost) of different novelty detection methods using the best selected feature sets (defined earlier in section 6.8.1).

Table 6.5: The performance measures of different classifiers using the best selected feature sets

	ND Performance measures (%)				ROC Plot	Time Cost
ND Methods	**B-Accuracy**	**F-Score**	**MCC**	**Gmean**	**AUC**	**(min)**
PARZEN	81.25	82.35	62.99	81.01	0.754	0.129
GMM	84.38	84.85	68.88	84.32	0.779	**0.062**
GPOC	81.25	80.00	62.99	81.01	0.779	1.057
SOM	84.38	82.76	69.99	83.85	0.768	4.306
Proposed	**87.50**	**86.67**	**75.59**	**87.28**	**0.863**	**0.127**

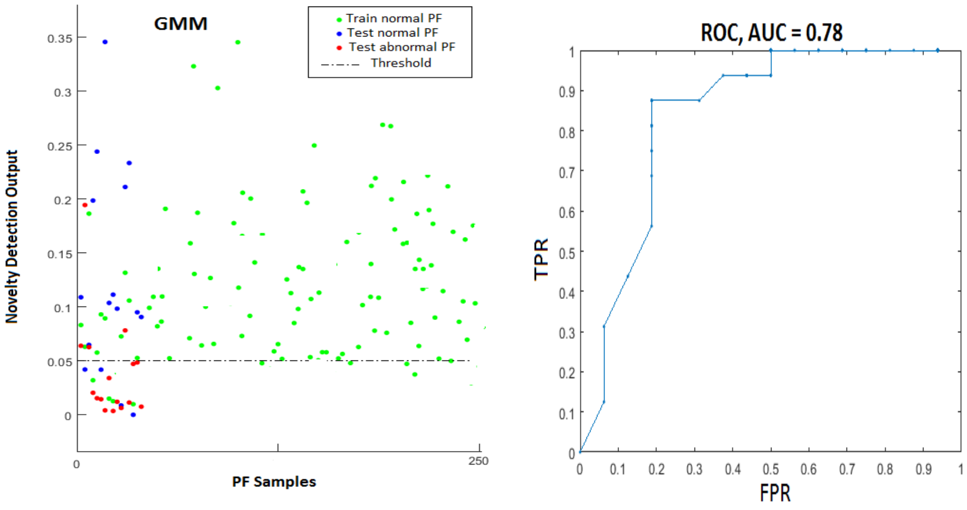
The results of all experiments suggested that proposed approach using SVDD module performed the best test performance results among all other novelty detection modules. Regarding other methods (Parzen, GMM, GPOC, SOM), the main deficiency is that they are sensitive to high dimensional PF datasets when just a small number of samples per class are used, due to the inaccurate threshold estimation problem. This deficiency is reflected in the low novelty performance measures shown in Table 6.5 for these techniques.

Figures 6.4, 6.5, 6.6, 6.7 and 6.8 illustrate the results of novelty detection methods output (scores) obtained using training normal data, testing normal data and testing abnormal data, and the ROC analysis using AUC for different novelty detection methods. The optimal threshold set by the validation PF dataset is shown by an horizontal dash line. Based on these novelty scores and the optimal thresholds a PF dataset point is categorized as abnormal if it exceeded the optimal threshold. Based on the previous novelty detection performance and the ROC curves analysis for all novelty detection methods the AUC was the best for the proposed approach using SVDD technique with 0.86, followed by GMM and GPOC with a score of 0.78, and SOM and Parzen with a score of 0.77 and 0.75, respectively. The analysis of the ROC curve for the SVDD approach confirms its high performance, as presented in Table 6.5 outputs of all channels. In terms of low AUC measure, it can be concluded that the PARZEN classifier impacts the most in the average misclassification outputs.



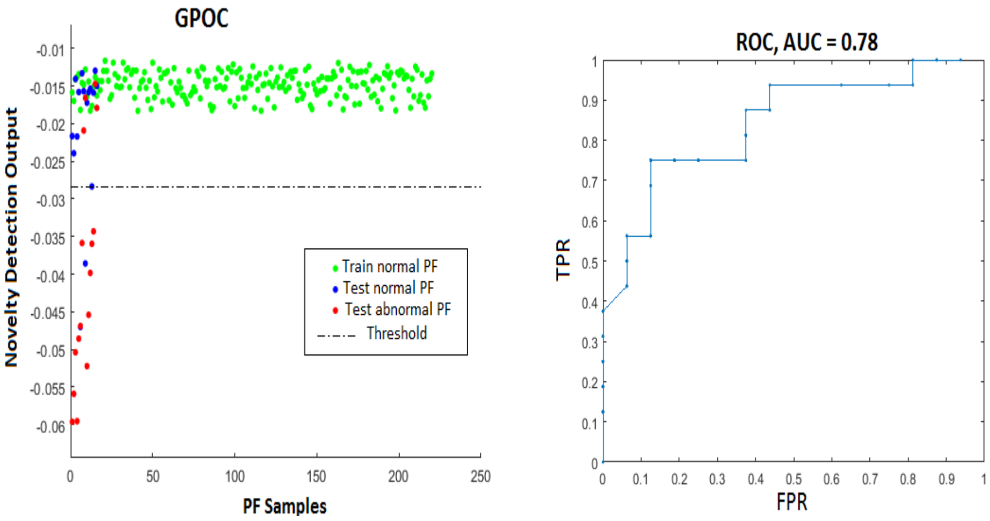
(a) **(b)**

Figure 6.4: Results of a Parzen Density window Estimator(PDE) for PF dataset. The threshold is set to 0.075, represented by a horizontal dash-and-dotted line. (a) Novelty scores (shown on y-axis) obtained using training normal data, testing normal and abnormal data; (d) ROC analysis for the Parzen method .The analysis of the ROC curve for the PARZEN class confirms its relatively low performance, as presented in Table 6.4. Normal data for training, normal data for testing, and abnormal data for testing are shown by green, blue and red, respectively.



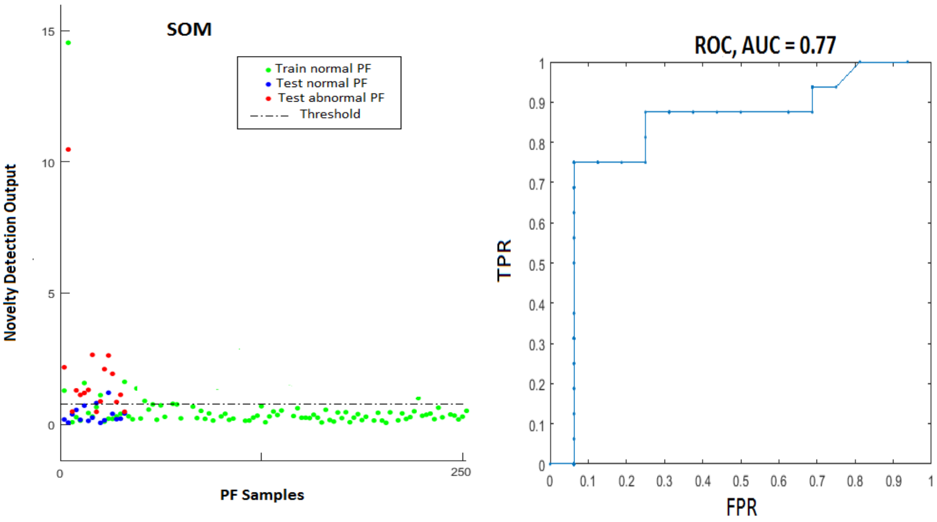
(a) **(b)**

Figure 6.5: Results of a Gaussian Mixture Models (GMM) window estimator for PF dataset. The optimal threshold is set to 0,05 and represented by a dash-and-dotted horizontal line.



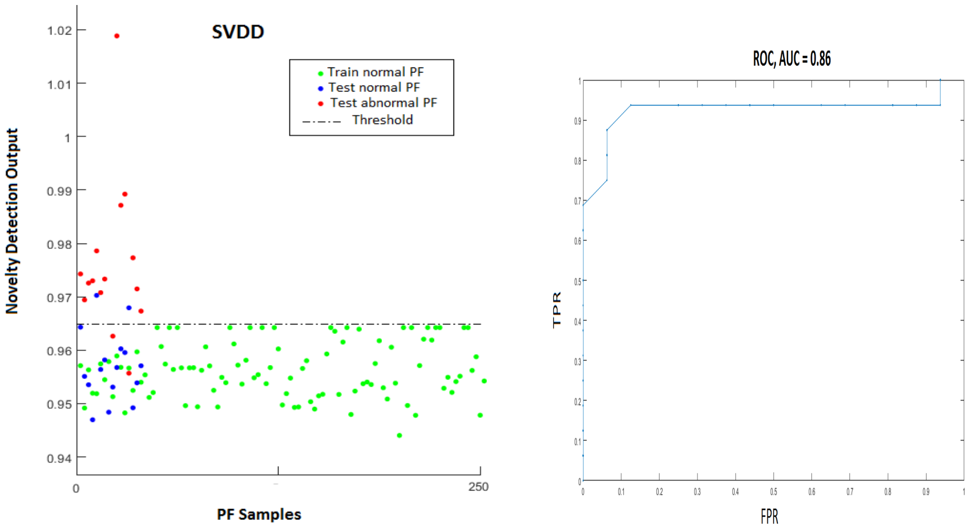
(a) (b)

Figure 6.6: Results of a GPOC model for PF dataset. The threshold is set to be -0.0027, represented by a horizontal line.



(a) (b)

Figure 6.7: Results of the SOM model for PF dataset. The threshold is set to be 0.8 and represented by a horizontal line.



(a) **(b)**

Figure 6.8: Results of a SVDD one-class classifier for PF dataset. The threshold is set to 0.965 and represented by a horizontal line. The analysis of the ROC curve for the SVDD approach confirms its high performance, as presented in Table 6.5

When considering the average classification execution time for different novelty detection methods (Table 6.5) using the best selected feature sets, the GMM model took up the least execution time with 0.06 min) followed by SVDD with (0.127 min), Parzen with (0.129 min), GPOC model with (1.057 min) and SOM with (4.129 min).

As reported in the the previous chapters, it is also clearly evident from the above results that the performance of most novelty detection models has been improved when using the best selected features. This also confirms the usefulness of using feature ranking and selection methods for improving the one-class classification performance and the execution time, and helps in selecting an appropriate novelty detection model for the one-class US PF classification approach. Using the novelty detection model analysis, it is also clearly evident from the overall results presented earlier in the previous section that SVDD outperforms other one-class classification modules and achieved its best in terms high performance measures.

6.9 Summary

Most traditional machine learning classification approaches to ultrasound anomaly detection rely on binary classification. They require symptomatic dataset for training, which is in some cases hard to acquire, and do not deal with the class imbalance between asymptomatic and symptomatic dataset. In contrast, this study has presented an attempt approach for US PF abnormality detection based on one-class SVDD classification module which creates a normality model based only on normal PF classes and declaring novelties or outliers to this normality class as symptomatic PF samples. The experimental results demonstrate its promising performance when compared to other selected modules. Interesting future directions related to this research are: (1) to automatically estimate the novelty detection threshold; (2) extend the proposed approach for multi-class classification with novelty detection for the classification of different symptomatic classes, plantar Fasciitis, plantar fascial tears, plantar fibromatosis; (3) the integration of 3-D PF representation, to provide a better representation of different PF structures and to improve the characterization and the discrimination of different PF subjects for different PF structures (rearfoot, midfoot and forefoot).

Chapter 7

Conclusions

7.1 Overview

This chapter brings the thesis to a close by drawing the conclusions resulting from the research work and approaches reported and investigated in the previous chapters to: (1) visually improve the PF US images; (2) segment the plantar fascia region using different structures and estimate their thicknesses; (3) characterize and distinguish between asymptomatic and symptomatic PF US subjects; (4) detect and analyse the plantar fascia abnormality behaviour (novelty or abnormality threshold) using novelty detection approach. The principal achievements and performances are also discussed along with the main recommendations for future scope.

7.2 Speckle reduction evaluation study

Ultrasound (US) imaging is a widely used and easy to use technology for medical plantar fascia diagnosis and prognosis, due to its relative safety (no-ionizing radiation), availability, portability, and lower cost, as compared to other imaging methods. It is crucial from the viewpoint of patients suffering from common disorders, because it has considerable potential for identifying suitable treatments for the related diseases. However, the main issue related to US imaging is the generation of speckled images, which might adversely affect medical image interpretation and diagnosis.

The main aim of this study is to perform a comparative evaluation of some existing speckle-reducing filtering methods (selected based on some previous research studies and medical experts) in the assessment of 2-D noisy plantar fascia ultrasound datasets. For this purpose, a medical image processing MATLAB program and GUI frame work was developed. This framework extends some MATLAB functionalities and has the capability to be combined with other medical image processing toolboxes. This frame work supports a wide range of ultrasound image despeckling functionalities as shown in Appendix B for the evaluation of seven groups and 16 despeckling filters. This includes, Median, adaptive local statistic filters (Mean Variance, Wiener), homogeneity, geometric, anisotropic diffusion, wavelet-based filtering and complex wavelet based filters using different thresholding methods (hard, soft, trimmed, bivariate) and hybrid filters such as DLWFDW and HybridMedian.

The trade-off between the acoustic speckle noise elimination and image detail preservation was analysed using three proposed image quality evaluation protocols. This includes: (1) 11 image qualitative metrics; (2) feature ranking and selection analysis; and (3) visual evaluation by two medical experts.

The results of this study indicate that the filters based on dual tree complex wavelet transform (DT-CWT) using BayesShrink subband thresholding and different thresholding functions namely, soft, hard, trimmed and bivariate ((DT-CWT_S, DT-CWT_H, DT-CWT_T and DT-CWT_B)) achieved the best results, followed by DPAD, DLWFDW filters. These filters have demonstrated the followings: (1) the ability to reduce speckle noise effectively while preserving and enhancing the edges of the PF US images, as compared to other filters in this study; (2) the ability to improve feature ranking and selection process and to define the most prominent feature sets; and (3) the ability to improve visual evaluation using two medical experts and produce visually more pleasing images.

The effectiveness of these filters have also proven that they are important mathematical tool which can have a great potential for PF US imaging segmentation, features extraction, selection, registration and classification. The careful selection of speckle reduction filters is very significant in the despeckling of the PF US images. However, further investigation is needed to test the performance of these filters on a larger US images dataset (normal and abnormal PF US images) using more experts and analyse its impact on medical image applications.

7.3 Plantar fascia segmentation and thickness estimation in ultrasound images

Despite the advantages of US imaging, images are difficult to interpret during medical assessment. This is partly due to the size and position of the PF in relation to the adjacent tissues. It is therefore a requirement to devise a system that allows better and easier interpretation of PF ultrasound images during diagnosis. This study proposes an automatic segmentation approach which for the first time extracts ultrasound data to estimate size across three sections of the PF (rearfoot, midfoot and forefoot). For this purpose, a medical image processing MATLAB program and GUI frame work was developed. This framework extends some MATLAB functionalities and has the capability to be combined with other medical image processing toolboxes. This frame work supports a wide range of ultrasound image functionalities such as US images despeckling, segmentation, thickness estimation, feature extraction, classification and novelty detection as shown in Appendix C. The proposed segmentation method uses radial basic function artificial neural network module (RBF-ANN) in order to classify small overlapping patches as belonging or not-belonging to the region of interest (ROI) of the PF tissue. Features ranking and selection techniques were performed as a post-processing step for features extraction to reduce the dimension and number of the extracted features. The trained RBF-ANN classifies the image overlapping patches into PF and non-PF tissue, and then it is used to segment the desired PF region in different PF structures. The PF thickness was calculated using two different methods: distance transformation and area-length calculation algorithms.

The statistical analysis results revealed that the area-length thickness estimation approach outperform the distance transformation approach in terms of significant positive pairing between the manual thickness estimation and the automatic assessment. Additionally, these results have also proven that there is a significant difference between different PF structures (forefoot, midfoot and rearfoot) and the thicknesses of PF subjects vary along the length of the foot (as reported in the literature). It is also concluded that the overall estimated PF thickness for all normal subjects using the area-length method ranged from approximately 1.94 mm-3.56 mm for the rearfoot section, 1.57 mm-2.01

mm for the midfoot section, and 1.11 mm-1.57 mm for the forefoot section. The implementation of such system is very helpful to assist the pathologist for early diagnosis and the detection of the PF associated medical problems. This also reduce the effect of many challenges that face the physicians and pathologist. These challenges include the time required by physicians for PF pathology diagnosis and the subjectivity that accompanies manual delineations and PF thickness measurements. The effectiveness of the proposed segmentation approach suggests that it has great potential for PF US imaging classification and novelty detection applications.

7.4 Plantar fascia characterization and classification, based on various supervised machine learning techniques for ultrasound images

Since the examination of the plantar fascia (PF) ultrasound (US) images is subjective and based on the visual perceptions and manual biometric measurements carried out by medical experts, US images feature extraction, characterization and classification have been widely introduced for improving the accuracy of medical assessment, reducing its subjective nature and the time required by physicians for PF pathology diagnosis. This study introduces an automated supervised classification approach which distinguishes between symptomatic and asymptomatic PF cases. Such an approach will facilitate the characterization and the classification of the plantar fascia area for the identification of patients with inferior heel pain at risk of plantar fasciitis.

Six feature sets were extracted from the segmented PF region. Additionally, features normalization, features ranking and selection analysis using an unsupervised infinity selection method were also introduced (to rank the extracted features, based on their weights importance) for the characterization and the classification of symptomatic and asymptomatic PF subjects. In the characterisation of asymptomatic and symptomatic US PF subjects only the top 28 features were selected. Additionally, the F-score measure was also introduced to define and compare the best features for different classifiers (Linear-SVM, Kernel-SVM, LDA, KNN, CART DT and RBF-NN) using different selected feature sets (1-40). These classifiers were investigated for the classification of the

PF ultrasound subjects using 10-folded cross-validation method.

The performance of all classification models was assessed using confusion matrix attributes and some derived performance measures including recall, specificity, balanced accuracy, precision, F-score, MCC, average score, area under curve and execution cost.

It is clearly evident from the overall results presented in this study that the RBF-NN outperforms other classification modules and achieved its best in terms of low misclassified PF instances, high performance measures for Recall, F-Score, MCC and AUC and average measure, with cheaper execution cost. This also has proven the effectiveness of the RBF-NN approach when introduced in the classification and the discrimination of different US PF subjects.

7.5 Novelty detection model for ultrasound images of the plantar fascia tissue

Detecting anomalies in US medical images is a challenging process. Therefore, a suitable model for novelty detection in PF US images is in high demand. In this study we propose a one-class classification model using SVDD novelty detection classifier applied only to normal PF US dataset (252 normal PF US images with insufficient 32 abnormal images used only for testing and evaluation) for defining novelty scores and thresholds to separate asymptomatic PF subjects from symptomatic ones. This study also aims to address the class imbalance problem between asymptomatic and symptomatic dataset. The scores were estimated using only the normal training PF subjects. For the testing and the evaluation task, both normal and abnormal PF subjects were used. The optimal score (threshold) is defined and set by the testing and the validation PF dataset. For normal PF US subjects, the dataset was split into 3 different classes 60% for training, 20% for validation and 20% for testing. On the other hand the abnormal PF subjects were split randomly into two classes: 50% for validation and 50% for testing. In order to define the best selected features for each novelty detection classifier (Parzen, GMM, GPOC, SOM and SVDD), G-mean measure was calculated using 28 feature sets. The best novelty detection classification Gmean results were achieved by SVDD using the the top 16 (highly ranked) features with a value of 0.873. The performance of the proposed model using SVDD one-class novelty model is evaluated and compared with other novelty detection methods (Parzen, GMM, GPOC and SOM) using five different performance measures namely, B-Accuracy, F-score, MCC percent, Gmean, ROC plot AUC and time cost using the best selected feature sets. The experimental results demonstrate its promising performance when compared to other selected modules (Parzen, GMM, GPOC and SOM) in terms of high performance measures. This also indicates the possibility of detecting early warning of plantar fasciitis or other PF anomalies using novelty scores estimation approach.

7.6 Contributions to Knowledge summary

The contribution to knowledge that has emerged for this study is the implementation of a MatLab based GUI system Appendices B and C, which is able:

1. To reduce speckle noise from different ultrasound images using a an evaluation study for selecting the best despeckling method.

2. To enhance different ultrasound images using a selected enhancement method.

3. To automatically and correctly segment the PF region and precisely estimates its thickness from US images.

4. To classify different PF US images to normal or abnormal subjects using supervised and novelty detection classification techniques

The novelty of the new prototype system is that it offers different significant improvements: (1) the ability of the system to automatically segment the PF region (in different Structures: rearfoot, midfoot and forefoot) from different US images and estimate its thickness; (2) the ability of the system to reduce the time required by physicians for PF pathology diagnosis; (3) the ability of the system to reduce the subjectivity that accompanies manual delineations and PF thickness measurements; and (4) the ability of the system to classify different PF US images to normal or abnormal subjects and draw a novelty score (threshold) to differentiate between symptomatic and asymptomatic PF subjects.

This study has demonstrated the advantages of artificial neural network (ANN) supervised classification techniques in medical imaging due to their learning capabilities for solving complicated tasks such as US segmentation and classification. In this study the RBF-NN supervised learning approach is advantageous over the traditional ones when dealing with speckled and poor US images. Additionally, the integration of SVDD technique in one-class novelty detection model demonstrates its promising performance when compared to other selected modules (Parzen, GMM, GPOC and SOM) in terms of high performance measures. This also indicates the possibility of detecting early warning of plantar fasciitis or other PF anomalies using novelty scores estimation approach.

In general computing terms, the implemented matlab based system (Appendices B and C) tend to provide a general framework for speckle noise reduction, feature extraction, selection, segmentation, classification, PF thickness estimation, novelty detection and possible generalization to a wider range of US images.

7.7 Future Work

7.7.1 Use of artificial PF dataset

This study was conducted only on small size of real US images dataset especially abnormal cases (36 images) with the lack of ground truth images, this will limit the performance estimates and the results generalization of the four studies carried out in this thesis (despeckling, segmentation and thickness estimation, supervised classification and novelty detection). To overcome these disadvantages and limitation, we propose a solution (well documented in the literature) as a future work to extend the existing methods by generating artificial PF US datasets for normal and abnormal samples. (1) For speckle noise reduction study, the artificial PF dataset (normal and abnormal samples) can be used in the establishment of the PF ground truth (speckle-noise free images) and in the generation of speckled PF images. This will facilitate and improve the quantitative, qualitative, and visual evaluation. (2) For novelty detection studies and due to lack of abnormal samples, the simulated PF dataset can be used independently in the validation and testing phases, while the normal real PF dataset are used only for training, this approach will help in the early identification of the PF anomalies with low false-positive estimation and also show an indication of efficiency when comparing different anomaly detection techniques. (3) Artificial PF dataset can also be introduced in two-class classification modules (e.g ANN) to perform role of novelty detection task. Both real asymptomatic and artificial symptomatic PF data are used in the training phase (Markou and Singh, 2006). Further investigation may be required for PF artificial dataset generation procedures and methods used.

7.7.2 Further use of texture features and other selected methods in 3D image representation

This study use only in 2D US image representation for processing normal and abnormal US images, this will alter other useful information, such as wider location, texture context, and volume, etc. The only way of preserving these useful information is by using 3D. Additionally, most physicians prefer a 3D representation of the medical images for diagnosis (Luboz et al., 2014), because it appears to be closer to the real world. Therefore, (1) the extension of the implemented approaches for processing other pathological PF tissues (e.g. rupture, fibromatosis, fibroma) in 3D US imaging domain is advantageous, (2) the integration of other texture feature measures and other selection methods using 3D multi-dimensional representation provides a better representation of different PF structures (rearfoot, midfoot and forefoot) and it improves the characterization and the discrimination of different PF subjects, (3) the integration of multi-class classification (e.g. plantar Fasciitis, plantar fascial tears, and plantar fibromatosis) using the proposed novelty detection approach or other better approaches to improve the one-class or multi-class novelty detection classification performance, (4) Following the success of the RBF-NN segmentation, RBF-NN classification and SVDD novelty detection approaches using 2D PF US images, the potential of these approaches could be investigated for other generated 3D US images and videos to estimate PF thickness and its volume.

Bibliography

Abbott, J. G., Thurstone, F., 1979. Acoustic speckle: Theory and experimental analysis. Ultrasonic Imaging 1 (4), 303–324.

Abd-Elmoniem, K. Z., Youssef, A.-B., Kadah, Y. M., 2002. Real-time speckle reduction and coherence enhancement in ultrasound imaging via nonlinear anisotropic diffusion. IEEE Transactions on Biomedical Engineering 49 (9), 997–1014.

Abe, S., 2010. Support Vector Machines for Pattern Classification. Advances in Computer Vision and Pattern Recognition. Springer London.

Achim, A., Bezerianos, A., Tsakalides, P., 2001. Novel bayesian multiscale method for speckle removal in medical ultrasound images. IEEE transactions on medical imaging 20 (8), 772–783.

Acton, Q., 2013. Pneumothorax: New Insights for the Healthcare Professional: 2013 Edition: ScholarlyBrief. ScholarlyEditions.

Aeyels, D., 1991. On the dynamic behavior of the novelty detector and the novelty filter. In: Analysis of controlled dynamical systems. Springer, pp. 1–10.

Agatonovic-Kustrin, S., Beresford, R., 2000. Basic concepts of artificial neural network (ann) modeling and its application in pharmaceutical research. Journal of Pharmaceutical and Biomedical Analysis 22 (5), 717 – 727.

Aggarwal, C., 2015. Data Classification: Algorithms and Applications. Chapman & Hall/CRC Data Mining and Knowledge Discovery Series. CRC Press.
URL https://books.google.co.uk/books?id=nwQZCwAAQBAJ

Ahmed, N., Natarajan, T., Rao, K. R., 1974. Discrete cosine transform. IEEE transactions on Computers 100 (1), 90–93.

Aja-Fernández, S., Alberola-López, C., 2006a. On the estimation of the coefficient of variation for anisotropic diffusion speckle filtering. IEEE Transactions on Image Processing 15 (9), 2694–2701.

Aja-Fernández, S., Alberola-López, C., 2006b. On the estimation of the coefficient of variation for anisotropic diffusion speckle filtering. IEEE Transactions on Image Processing 15 (9), 2694–2701.

Akfirat, M., Sen, C., Günes, T., 2003. Ultrasonographic appearance of the plantar fasciitis. Clinical Imaging 27 (5), 353–357.

Albrecht, S., Busch, J., Kloppenburg, M., Metze, F., Tavan, P., 2000. Generalized radial basis function networks for classification and novelty detection: self-organization of optimal bayesian decision. Neural Networks 13 (10), 1075–1093.

Ali, S., Burge, R., 1988. New automatic techniques for smoothing and segmenting sar images. Signal Processing 14 (4), 335–346.

Almen, A., Tingberg, A., Mattsson, S., Besjakov, J., Kheddache, S., Lanhede, B., Månsson, L., Zankl, M., 2000. The influence of different technique factors on image quality of lumbar spine radiographs as evaluated by established cec image criteria. the British journal of radiology 73 (875), 1192–1199.

Amadasun, M., King, R., 1989. Textural features corresponding to textural properties. IEEE Transactions on systems, man, and Cybernetics 19 (5), 1264–1274.

Andria, G., Attivissimo, F., Lanzolla, A., Savino, M., Aug 2013. A suitable threshold for speckle reduction in ultrasound images. IEEE Transactions on Instrumentation and Measurement 62 (8), 2270–2279.

Angin, S., Crofts, G., Mickle, K. J., Nester, C. J., 2014. Posture Ultrasound evaluation of foot muscles and plantar fascia in pes planus. Gait & posture 40, 48–52.

Anitha, J., Vijila, C., Hemanth, D. J., 2009. An enhanced counter propagation neural network for abnormal retinal image classification. In: World Congress on Nature and Biologically Inspired Computing, NaBIC. IEEE, pp. 1–6.

Antonie, M.-L., Zaiane, O. R., Coman, A., 2001. Application of data mining techniques for medical image classification. In: Proceedings of the Second International Workshop on Multimedia Data Mining, MDM/KDD'2001, August 26th, 2001, San Francisco, CA, USA. pp. 94–101.

Augusteijn, M., Folkert, B., 2002. Neural network classification and novelty detection. International Journal of Remote Sensing 23 (14), 2891–2902.

Baeck, T., Fogel, D., Michalewicz, Z., 2000. Evolutionary Computation 1: Basic Algorithms and Operators. Basic algorithms and operators. Taylor & Francis.

Baradez, M.-O., McGuckin, C. P., Forraz, N., Pettengell, R., Hoppe, A., 2004. Robust and automated unimodal histogram thresholding and potential applications. Pattern Recognition 37 (6), 1131–1148.

Basu, S., Bilenko, M., Mooney, R. J., 2004. A probabilistic framework for semi-supervised clustering. In: Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp. 59–68.

Beeson, P., 2014. Plantar fasciopathy: revisiting the risk factors. Foot and Ankle Surgery 20 (3), 160–165.

Bezdek, J. C., 1981. Pattern Recognition with Fuzzy Objective Function Algorithms. Kluwer Academic Publishers, Norwell, MA, USA.

Bezdek, J. C., 2013. Pattern recognition with fuzzy objective function algorithms. Springer Science & Business Media.

Biradar, N., Dewal, M. L., Rohit, M. K., 2015. Speckle noise reduction in b-mode echocardiographic images: A comparison. IETE Technical Review 32 (6), 435–453.

Bishop, C. M., 1994. Novelty detection and neural network validation. IEE Proceedings-Vision, Image and Signal processing 141 (4), 217–222.

Bishop, C. M., 2006. Pattern recognition and machine learning. springer.

Blum, H., 1967. A Transformation for Extracting New Descriptors of Shape. In: Wathen-Dunn, W. (Ed.), Models for the Perception of Speech and Visual Form. MIT Press, Cambridge, pp. 362–380.

Borhani, M., Sedghi, V., Nayebi, M., Oct 2005. A new technique in passive coherent radar signal processing. In: Radar Conference, 2005. EURAD 2005. European. pp. 149–151.

Boriah, S., Chandola, V., Kumar, V., 2008. Similarity measures for categorical data: A comparative evaluation. In: Proceedings of the 2008 SIAM International Conference on Data Mining. SIAM, pp. 243–254.

Borş, A. G., Pitas, I., 1996. Median radial basis function neural network. IEEE Transactions on Neural Networks 7 (6), 1351–1364.

Borş, A. G., Pitas, I., 1999. Object classification in 3-d images using alpha-trimmed mean radial basis function network. IEEE Transactions on Image Processing 8 (12), 1744–1756.

Boussouar, A., Meziane, F., 2018a. Computer-based medical ultrasound image processing system and methods.

Boussouar, A., Meziane, F., 2018b. Novelty detection for ultrasound images of the plantar fascia.

Boussouar, A., Meziane, F., 2018c. Plantar fascia characterization and classification based on machine learning techniques for ultrasound images. Submitted to Springer Neural Computing and Applications.

Boussouar, A., Meziane, F., Crofts, G., 2017a. Plantar fascia segmentation and thickness estimation in ultrasound images. Computerized Medical Imaging and Graphics 56, 60–73.

Boussouar, A., Meziane, F., Hogg, P., Hashmi, F., 2017b. Speckle noise reduction in ultrasound imaging of the plantar fascia, a comparative evaluation. Submitted to ACM Computing Surveys (Second round review).

Bovik, A., 2010. Handbook of Image and Video Processing. Communications, Networking and Multimedia. Elsevier Science.

Bradley, A. P., 1997. The use of the area under the roc curve in the evaluation of machine learning algorithms. Pattern recognition 30 (7), 1145–1159.

Breaban, M., Luchian, H., 2012. Outlier detection with nonlinear projection pursuit. International Journal of Computers Communications & Control 8 (1), 30–36.

Breiman, L., Friedman, J., Stone, C. J., Olshen, R. A., 1984. Classification and regression trees. CRC press.

Brigham, E. O., Brigham, E. O., Rey Pastor, J., Pastor, R., Apostol, T. M. T. M., Rodríguez, M., Rodríguez, M. R., Martínez, M. R., Edwards, C. H., Edwards, D. E. H., et al., 1988. The fast Fourier transform and its applications. No. 517.443. Prentice Hall,.

Broomhead, D. S., Lowe, D., 1988. Radial basis functions, multi-variable functional interpolation and adaptive networks. Tech. rep., DTIC Document.

Buchbinder, R., 2004. Plantar fasciitis. New England Journal of Medicine 350 (21), 2159–2166.

Buie, H. R., Campbell, G. M., Klinck, R. J., MacNeil, J. A., Boyd, S. K., 2007. Automatic segmentation of cortical and trabecular compartments based on a dual threshold technique for in vivo micro-ct bone analysis. Bone 41 (4), 505–515.

Burckhardt, C., Jan 1978. Speckle in ultrasound b-mode scans. Sonics and Ultrasonics, IEEE transactions on medical imaging 25 (1), 1–6.

Busse, L., Crimmins, T., Fienup, J., Nov 1995. A model based approach to improve the performance of the geometric filtering speckle reduction algorithm. In: Ultrasonics Symposium, 1995. Proceedings., 1995 IEEE. Vol. 2. pp. 1353–1356.

Cai-lian, L., Ji-xiang, S., Yao-hong, K., Sept 2010. Adaptive image denoising by a new thresholding function. In: Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on. pp. 1–5.

Campbell, C., Bennett, K. P., 2001. A linear programming approach to novelty detection. Advances in neural information processing systems, 395–401.

Campilho, A., Karray, F., 2016. Image Analysis and Recognition: 13th International Conference, ICIAR 2016, in Memory of Mohamed Kamel, Póvoa de Varzim, Portugal, July 13-15, 2016, Proceedings. Vol. 9730. Springer.

Cantone, D., Ferro, A., Pulvirenti, A., Recupero, D. R., Shasha, D., 2005. Antipole tree indexing to support range search and k-nearest neighbor search in metric spaces. IEEE Transactions on Knowledge and Data Engineering 17 (4), 535–550.

Cha, I., Kassam, S., Jun 1996. Rbfn restoration of non-linearly degraded images. IEEE Transactions on Image Processing 5 (6), 964–975.

Chambolle, A., De Vore, R., Lee, N.-Y., Lucier, B., March 1998. Nonlinear wavelet image processing: variational problems, compression, and noise removal through wavelet shrinkage. IEEE Transactions on Image Processing 7 (3), 319–335.

Chan, T., Marquina, A., Mulet, P., 2000. High-order total variation-based image restoration. SIAM Journal on Scientific Computing 22 (2), 503–516.

Chan, T., Vese, L., Feb 2001. Active contours without edges. IEEE Transactions on Image Processing 10 (2), 266–277.

Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: A survey. ACM computing surveys (CSUR) 41 (3), 15.

Chandola, V., Boriah, S., Kumar, V., 2008. Understanding categorical similarity measures for outlier detection. Technical report 08–008, University of Minnesota.

Chang, C.-y., Member, S., Lei, Y.-f., Tseng, C.-h., Shih, S.-r., 2010. Thyroid Segmentation and Volume Estimation in Ultrasound Images 57 (6), 1348–1357.

Chang, L., Lie, W., 2006. Advances in Image and Video Technology: First Pacific Rim Symposium, PSIVT 2006, Hsinchu, Taiwan, December 10-13, 2006, Proceedings. Image Processing, Computer Vision, Pattern Recognition, and Graphics. Springer.

Chang, R., 2010. Plantar fasciitis: Biomechanics, atrophy and muscle energetics. PhD dissertation, University of Massachusetts Amherst, USA.

Chang, S., Yu, B., Vetterli, M., Sep 2000. Adaptive wavelet thresholding for image denoising and compression. IEEE Transactions on Image Processing 9 (9), 1532–1546.

Chang, Y.-L., Li, X., 1994. Adaptive image region-growing. Image Processing, IEEE Transactions on 3 (6), 868–872.

Chen, C., 2015. Handbook of Pattern Recognition and Computer Vision. World Scientific Publishing Company.

Chen, C.-h., Pau, L.-F., Wang, P. S.-p., 2010. Handbook of pattern recognition and computer vision. Vol. 27. World Scientific.

Chen, G., Qian, S.-E., March 2011. Denoising of hyperspectral imagery using principal component analysis and wavelet shrinkage. IEEE Transactions on Geoscience and Remote Sensing 49 (3), 973–980.

Chen, G., Zhu, W.-P., 2008. Image denoising using three scales of wavelet coefficients. In: Sun, F., Zhang, J., Tan, Y., Cao, J., Yu, W. (Eds.), Advances in Neural Networks - ISNN 2008. Vol. 5264 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 376–383.

Chen, S., Grant, P., Cowan, C., Nov 1991. Orthogonal least squares algorithm for training multi-output radial basis function networks. In: Artificial Neural Networks, 1991., Second International Conference on. pp. 336–339.

Chen, Y., Lei, L., Ji, Z.-C., Sun, J.-F., 2007. Adaptive wavelet threshold for image denoising by exploiting inter-scale dependency. In: Huang, D.-S., Heutte, L., Loog, M. (Eds.), Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues. Vol. 4681 of Lecture Notes in Computer Science. Springer Berlin Heidelberg, pp. 869–878.

Cheng, J.-W., Tsai, W.-C., Yu, T.-Y., Huang, K.-Y., 2012. Reproducibility of sonographic measurement of thickness and echogenicity of the plantar fascia. Journal of Clinical Ultrasound 40 (1), 14–19.

Chitre, Y., Dhawan, A. P., Moskowitz, M., 1994. Artificial neural network based classification of mammographic microcalcifications using image structure features. State of the Art in Digital Mammographic Image Analysis, Boyer, KW, et al., editors, 167–197.

Christodoulou, C., Loizou, C., Pattichis, C., Pantziaris, M., Kyriakou, E., Pattichis, M., Schizas, C., Nicolaides, A., 2002. De-speckle filtering in ultrasound imaging of the carotid artery. In: Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society EMB-S/BMES Conference, 2002. Proceedings of the Second Joint. Vol. 2. pp. 1027–1028.

Christodoulou, C. I., Pattichis, C. S., Pantziaris, M., Nicolaides, A., 2003. Texture-based classification of atherosclerotic carotid plaques. IEEE transactions on medical imaging 22 (7), 902–912.

Chun, Y. D., Seo, S. Y., Kim, N. C., Sept 2003. Image retrieval using bdip and bvlc moments. IEEE Transactions on Circuits and Systems for Video Technology 13 (9), 951–957.

Clement, D., Taunton, J., Smart, G., McNicol, K., 1981. A survey of overuse running injuries. Medicine & Science in Sports & Exercise 13 (2), 83.

Clifton, D. A., Clifton, L. A., Bannister, P. R., Tarassenko, L., 2008. Automated novelty detection in industrial systems. In: Advances of Computational Intelligence in Industrial Systems. Springer, pp. 269–296.

Compton, P., Cao, T. M., 2006. Evaluation of incremental knowledge acquisition with simulated experts. In: Australasian Joint Conference on Artificial Intelligence. Springer, pp. 39–48.

Cooley, J. W., Tukey, J. W., 1965. An algorithm for the machine calculation of complex fourier series. Mathematics of computation 19 (90), 297–301.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine learning 20 (3), 273–297.

Crimmins, T. R., May 1985. Geometric filter for speckle reduction. Appl. Opt. 24 (10), 1438–1443.

Crofts, G., Angin, S., Mickle, K. J., Hill, S., Nester, C. J., 2014. Reliability of ultrasound for measurement of selected foot structures. Gait & posture 39 (1), 35–9.

De Stefano, C., Sansone, C., Vento, M., 2000. To reject or not to reject: that is the question-an answer in case of neural classifiers. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews) 30 (1), 84–94.

DeKruger, D., Hunt, B. R., 1994. Image processing and neural networks for recognition of cartographic area features. Pattern Recognition 27 (4), 461–483.

Deserno, T., 2011. Biomedical Image Processing. Biological and Medical Physics, Biomedical Engineering. Springer Berlin Heidelberg.

Deshpande, R., Ramalingam, R., Chockalingam, N., Naemi, R., Branthwaite, H., Sundar, L., April 2013. An automated segmentation technique for the processing of foot ultrasound images. In: IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing. pp. 380–383.

Dey, N., Ashour, A., Borra, S., 2017. Classification in BioApps: Automation of Decision Making. Lecture Notes in Computational Vision and Biomechanics. Springer International Publishing.

Dhawan, A., 2011. Medical Image Analysis. IEEE Press Series on Biomedical Engineering. Wiley.

Di Stefano, L., Bulgarelli, A., 1999. A simple and efficient connected components labeling algorithm. In: IEEE International Conference on Image Analysis and Processing. Proceedings. IEEE, pp. 322–327.

Dice, L. R., 1945. Measures of the amount of ecologic association between species. Ecology 26 (3), pp. 297–302.

Do, M. N., Vetterli, M., 2005. The contourlet transform: an efficient directional multiresolution image representation. IEEE Transactions on image processing 14 (12), 2091–2106.

Donoho, D., May 1995. De-noising by soft-thresholding. IEEE Transactions on Information Theory 41 (3), 613–627.

Donoho, D. L., Johnstone, I. M., 1995. Adapting to unknown smoothness via wavelet shrinkage. Journal of the american statistical association 90 (432), 1200–1224.

Donoho, D. L., Johnstone, J. M., 1994. Ideal spatial adaptation by wavelet shrinkage. Biometrika 81 (3), 425–455.

Dougherty, G., 2012. Pattern Recognition and Classification: An Introduction. SpringerLink : Bücher. Springer New York.

Duda, R., Hart, P., Stork, D., 2001. Pattern classification. Pattern Classification and Scene Analysis: Pattern Classification. Wiley.

Duda, R. O., Hart, P. E., et al., 1973. Pattern classification and scene analysis. Vol. 3. Wiley New York.

Fabrikant, J. M., Park, T. S., 2011. Plantar fasciitis (fasciosis) treatment outcome study: Plantar fascia thickness measured by ultrasound and correlated with patient self-reported improvement. The Foot 21 (2), 79 – 83.

Fahlman, S. E., Lebiere, C., 1990. The cascade-correlation learning architecture.

Fang, H.-T., Huang, D.-S., 2004. Wavelet de-noising by means of trimmed thresholding. In: Intelligent Control and Automation, 2004. WCICA 2004. Fifth World Congress on. Vol. 2. IEEE, pp. 1621–1624.

Fawcett, T., 2005. An introduction to roc analysis. 2164 staunton court. Palo Alto: Institute for the Study of Learning and Expertise.

Finn, S., Glavin, M., Jones, E., January 2011. Echocardiographic speckle reduction comparison. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control 58 (1), 82–101.

Fisher, R. A., 1936. The use of multiple measurements in taxonomic problems. Annals of human genetics 7 (2), 179–188.

Fix, E., Hodges Jr, J. L., 1951. Discriminatory analysis-nonparametric discrimination: consistency properties. Tech. rep., DTIC Document.

Fornage, B. D., 1993. Ultrasound of the breast. Ultrasound Quart 11 (1), 1–40.

Foster, D., Arditi, M., Foster, F., Patterson, M., Hunt, J., 1983. Computer simulations of speckle in b-scan images. Ultrasonic Imaging 5 (4), 308 – 330.

Freeman, W. T., Adelson, E. H., 1991. The design and use of steerable filters. IEEE Transactions on Pattern Analysis & Machine Intelligence (9), 891–906.

Frost, V. S., Stiles, J. A., Shanmugan, K., Holtzman, J., March 1982. A model for radar images and its application to adaptive digital filtering of multiplicative noise. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-4 (2), 157–166.

Fu, K., Mui, J., 1981. A survey on image segmentation. Pattern Recognition 13 (1), 3 – 16.

Ganzalez, R. C., Woods, R. E., 2002. Digital Image Processing, 2nd Edition. Prentice Hall.

Gao, X., Xie, W., 2000. Advances in theory and applications of fuzzy clustering. Chinese Science Bulletin 45 (11), 961–970.

Gao, X.-b., 2004. Fuzzy cluster analysis and its applications. Xian: Xidian University Publishing House.

Gardner, A. B., Krieger, A. M., Vachtsevanos, G., Litt, B., 2006. One-class novelty detection for seizure analysis from intracranial eeg. Journal of Machine Learning Research 7 (Jun), 1025–1044.

Gonzalez, R., Wintz, P., 1977. Digital image processing.

Gonzalez, R., Woods, R., 2011. Digital Image Processing. Pearson Education.

Gonzalez, R., Woods, R., Eddins, S., 2010. Digital Image Processing Using MATLAB. Tata McGraw Hill Education.

Goodman, J. W., 1976. Some fundamental properties of speckle. JOSA 66 (11), 1145–1150.

Greer, J. B., Bertozzi, A. L., 2004. Traveling wave solutions of fourth order pdes for image processing. SIAM Journal on Mathematical Analysis 36 (1), 38–68.

Grinblat, G. L., Izetta, J., Granitto, P. M., 2010. Svm based feature selection: Why are we using the dual? In: Ibero-American Conference on Artificial Intelligence. Springer, pp. 413–422.

Grohman, W. M., Dhawan, A. P., 2001. Fuzzy convex set-based pattern classification for analysis of mammographic microcalcifications. Pattern Recognition 34 (7), 1469–1482.

Guo, G., Wang, H., Bell, D., Bi, Y., Greer, K., 2003. Knn model-based approach in classification. In: On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE. Springer, pp. 986–996.

Gupta, S., Chauhan, R., Saxena, S., 2005a. Homomorphic wavelet thresholding technique for denoising medical ultrasound images. Journal of medical engineering & technology 29 (5), 208–214.

Gupta, S., Chauhan, R., Saxena, S., 2005b. Robust non-homomorphic approach for speckle reduction in medical ultrasound images. Medical and Biological Engineering and Computing 43 (2), 189–195.

Gupta, S., Kaur, L., Chauhan, R., Saxena, S., 2007. A versatile technique for visual enhancement of medical ultrasound images. Digital Signal Processing 17 (3), 542–560.

Guyon, I., Weston, J., Barnhill, S., Vapnik, V., 2002. Gene selection for cancer classification using support vector machines. Machine learning 46 (1), 389–422.

Ham, F. M., Kostanic, I., 2000. Principles of Neurocomputing for Science and Engineering, 1st Edition. McGraw-Hill Higher Education.

Han, J., Pei, J., Kamber, M., 2006. Data Mining, Southeast Asia Edition. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science.

Haralick, R. M., 1979. Statistical and structural approaches to texture. Proceedings of the IEEE 67 (5), 786–804.

Haralick, R. M., Shanmugam, K., Dinstein, I. H., 1973. Textural features for image classification. IEEE Transactions on Systems, Man and Cybernetics SMC-3 (6), 610–621.

Haring, S., Viergever, M. A., Kok, J. N., 1994. Kohonen networks for multiscale image segmentation. Image and vision computing 12 (6), 339–344.

Heimann, T., Van Ginneken, B., Styner, M., Arzhaeva, Y., Aurich, V., Bauer, C., Beck, A., Becker, C., Beichel, R., Bekes, G., et al., 2009. Comparison and evaluation of methods for liver segmentation from ct datasets. IEEE Transactions on Medical Imaging 28 (8), 1251–1265.

Hiremath, P., Akkasaligar, P. T., Badiger, S., 2011. Speckle noise reduction in medical ultrasound images. Advancements and Breakthroughs in Ultrasound Imaging. InTech.

Ho, T., Rouat, J., 1997. A novelty detector using a network of integrate and fire neurons. Artificial Neural NetworksICANN'97, 103–108.

Hogg, P., Blindell, P., 2012. Software for image quality evaluation using a forced choice method. In: United Kingdom radiological conference. British Institute of Radiology, Manchester/London, UK, p. 139.

Horng, M.-H., 2009. Multi-class support vector machine for classification of the ultrasonic images of supraspinatus. Expert Systems with Applications 36 (4), 8124–8133.

Horng, M.-H., 2010. Performance evaluation of multiple classification of the ultrasonic supraspinatus images by using ml, rbfnn and svm classifiers. Expert Systems with Applications 37 (6), 4146–4155.

Horng, M.-H., 2013. The glowworm swarm optimization for training the radial basis function network in ultrasonic supraspinatus image classification. Advanced Science Letters 19 (9), 2724–2727.

Hough, P. V., 1962. Method and means for recognizing complex patterns. Tech. rep.

Huang, C.-K., Kitaoka, H. B., An, K.-N., Chao, E. Y., 1993. Biomechanical evaluation of longitudinal arch stability. Foot & Ankle International 14 (6), 353–357.

Hum, Y., 2013. Segmentation of Hand Bone for Bone Age Assessment. SpringerBriefs in Applied Sciences and Technology. Springer Singapore.

Huo, Z., Giger, M. L., Vyborny, C. J., 2001. Computerized analysis of multiple-mammographic views: Potential usefulness of special view mammograms in computer-aided diagnosis. IEEE Transactions on Medical Imaging 20 (12), 1285–1292.

Insana, M. F., Hall, T. J., Cox, G. G., Rosenthal, S. J., 1989. Progress in quantitative ultrasonic imaging. Vol. 1090. pp. 2–9.

Ioannidis, A., Kazakos, D., Watson, D. D., 1984. Application of median filtering on nuclear medicine scintigram images. In: Proc. 7th Int. Conf. Pattern Recognition. Vol. 33.

Jackson, I., of Cambridge. Department of Applied Mathematics, U., Phyiscs, T., 1988. Radial Basis Functions: A Survey and New Results. University of Cambridge Department of Applied Mathematics and Theoretical Physics.

Jagota, A., 1991. Novelty detection on a very large number of memories stored in a hopfield-style network. In: Neural Networks, 1991., IJCNN-91-Seattle International Joint Conference on. Vol. 2. IEEE, pp. 905–vol.

Jain, A., 1989. Fundamentals of Digital Image Processing. Prentice-Hall information and system sciences series. Prentice Hall.

Jain, A. K., Dubes, R. C., 1988. Algorithms for clustering data. Prentice-Hall, Inc.

Jain, P., Tyagi, V., 2014. A survey of edge-preserving image denoising methods. Information Systems Frontiers, 1–12.

Japkowicz, N., Stephen, S., Oct. 2002. The class imbalance problem: A systematic study. Intell. Data Anal. 6 (5), 429–449.

Jin, Y., Angelini, E., Laine, A., 2005. Wavelets in medical image processing: denoising, segmentation, and registration. In: Handbook of biomedical image analysis. Springer, pp. 305–358.

Jolliffe, I., 2002. Principal component analysis. Wiley Online Library.

Kang, C.-C., Wang, W.-J., Kang, C.-H., 2012. Image segmentation with complicated background by using seeded region growing. AEU-International Journal of Electronics and Communications 66 (9), 767–771.

Kang, J., Lee, J. Y., Yoo, Y., June 2016. A new feature-enhanced speckle reduction method based on multiscale analysis for ultrasound b-mode imaging. IEEE Transactions on Biomedical Engineering 63 (6), 1178–1191.

Kang, W.-X., Yang, Q.-Q., Liang, R.-P., 2009. The comparative research on image segmentation algorithms. 2009 First International Workshop on Education Technology and Computer Science, 703–707.

Kass, M., Witkin, A., Terzopoulos, D., 1988. Snakes: Active contour models. International journal of computer vision 1 (4), 321–331.

Kelikian, A., 2012. Sarrafian's Anatomy of the Foot and Ankle: Descriptive, Topographic, Functional. Wolters Kluwer Health.

Kemmler, M., Rodner, E., Denzler, J., 2010. One-class classification with gaussian processes. In: Asian Conference on Computer Vision. Springer, pp. 489–500.

Kennedy, K., Mac Namee, B., Delany, S. J., 2009. Learning without default: A study of one-class classification and the low-default portfolio problem. In: Irish Conference on Artificial Intelligence and Cognitive Science. Springer, pp. 174–187.

Kim, J.-Y., Kim, L.-S., Hwang, S.-H., 2001. An advanced contrast enhancement using partially overlapped sub-block histogram equalization. IEEE transactions on circuits and systems for video technology 11 (4), 475–484.

Kim, K., Joukov, N., 2016. Information Science and Applications (ICISA) 2016. Lecture Notes in Electrical Engineering. Springer Singapore.

Kim, T. K., Paik, J. K., Kang, B. S., 1998. Contrast enhancement system using spatially adaptive histogram equalization with temporal filtering. IEEE Transactions on Consumer Electronics 44 (1), 82–87.

Kim, Y.-T., 1997. Contrast enhancement using brightness preserving bi-histogram equalization. IEEE transactions on Consumer Electronics 43 (1), 1–8.

King, S., King, D., Astley, K., Tarassenko, L., Hayton, P., Utete, S., 2002. The use of novelty detection techniques for monitoring high-integrity plant. In: Control Applications, 2002. Proceedings of the 2002 International Conference on. Vol. 1. IEEE, pp. 221–226.

Kingsbury, N., Mar 1999. Shift invariant properties of the dual-tree complex wavelet transform. In: Acoustics, Speech, and Signal Processing, 1999. Proceedings., 1999 IEEE International Conference on. Vol. 3.

Kingsbury, N. G., 1998. The dual-tree complex wavelet transform: a new technique for shift invariance and directional filters. In: Proc. 8th IEEE DSP Workshop, Utah. Vol. 8. Citeseer, p. 86.

Kohonen, T., 1982. Self-organized formation of topologically correct feature maps. Biological cybernetics 43 (1), 59–69.

Kohonen, T., 1990. The self-organizing map. Proceedings of the IEEE 78 (9), 1464–1480.

Kondo, T., Ueno, J., Takao, S., Dec 2011. Hybrid gmdh-type neural network using artificial intelligence and its application to medical image diagnosis of liver cancer. In: System Integration (SII), 2011 IEEE/SICE International Symposium on. pp. 1101–1106.

Kotsiantis, S. B., 2007. Supervised machine learning: a review of classification techniques. Informatica (03505596) 31 (3).

Kovacevic, D., Loncaric, S., 1997. Radial basis function-based image segmentation using a receptive field. In: Proceedings of the 10th IEEE Symposium on Computer-Based Medical Systems (CBMS '97). CBMS '97. IEEE Computer Society, Washington, DC, USA, pp. 126–.

Kremkau, F., Forsberg, F., 2015. Sonography Principles and Instruments, 9th Edition. Elsevier - Health Sciences Division.

Krissian, K., Westin, C.-F., Kikinis, R., Vosburgh, K. G., 2007. Oriented speckle reducing anisotropic diffusion. IEEE Transactions on Image Processing 16 (5), 1412–1424.

Kuan, D. T., Sawchuk, A., Strand, T. C., Chavel, P., Mar 1987. Adaptive restoration of images with speckle. IEEE Transactions on Acoustics, Speech and Signal Processing 35 (3), 373–383.

Kubat, M., Holte, R. C., Matwin, S., 1998. Machine learning for the detection of oil spills in satellite radar images. Machine learning 30 (2-3), 195–215.

Kwong, P., Kay, D., Voner, R., White, M., January 1988. Plantar fasciitis. mechanics and pathomechanics of treatment. Clinics in sports medicine 7 (1), 119126.

Lai, K., Dewi, D., 2015. Medical Imaging Technology: Reviews and Computational Applications. Lecture Notes in Bioengineering. Springer Singapore.

Laine, A., Fan, J., 1993. Texture classification by wavelet packet signatures. IEEE Transactions on pattern analysis and machine intelligence 15 (11), 1186–1191.

Lal, S., Chandra, M., Upadhyay, G. K., 2009. Noise removal algorithm for images corrupted by additive gaussian noise. International Journal of Recent Trends in Engineering 2 (1), 199–206.

Landorf, K. B., Keenan, A.-M., Herbert, R. D., 2006. Effectiveness of foot orthoses to treat plantar fasciitis: a randomized trial. Archives of internal medicine 166 (12), 1305–1310.

Lankton, S., Tannenbaum, A., Nov. 2008. Localizing region-based active contours. IEEE Transactions on Image Processing 17 (11), 1–11.

Laws, K. I., 1980. Rapid texture identification. Vol. 0238. pp. 0238 – 0238 – 6.

Lee, J.-S., March 1980. Digital image enhancement and noise filtering by use of local statistics. Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-2 (2), 165–168.

Lee, J.-S., 1981a. Refined filtering of image noise using local statistics. Computer graphics and image processing 15 (4), 380–389.

Lee, J.-S., 1981b. Speckle analysis and smoothing of synthetic aperture radar images. Computer Graphics and Image Processing 17 (1), 24–32.

Lee, M.-S., Chen, M.-Y., Yen, C.-L., 2013. Quantitative comparison of speckle smoothing for ultrasound images using besov norm. Current Bioinformatics 8 (1), 25–34.

Lefebvre, F., Berger, G., Laugier, P., Feb 1998. Automatic detection of the boundary of the calcaneus from ultrasound parametric images using an active contour model; clinical assessment. IEEE transactions on medical imaging 17 (1), 45–52.

Legendre, P., Legendre, L., 2012. Numerical Ecology. Developments in Environmental Modelling. Elsevier Science.

Lendaris, G. G., Stanley, G. L., 1970. Diffraction-pattern sampling for automatic pattern recognition. Proceedings of the IEEE 58 (2), 198–216.

Lim, T.-S., Loh, W.-Y., Shih, Y.-S., 2000. A comparison of prediction accuracy, complexity, and training time of thirty-three old and new classification algorithms. Machine learning 40 (3), 203–228.

Lin, Y., Cai, J., March 2010. A new threshold function for signal denoising based on wavelet transform. In: Measuring Technology and Mechatronics Automation (ICMTMA), 2010 International Conference on. Vol. 1. pp. 200–203.

Liu, D., Xie, S., Li, Y., Zhao, D., El-Alfy, E., 2018. Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, November 14–18, 2017, Proceedings. No. pt. 5. Springer International Publishing.

Liu, H., Motoda, H., 2007. Computational methods of feature selection. CRC Press.

Liu, J., Gao, F., Li, Z., July 2011. A model of image denoising based on partial differential equations. In: Multimedia Technology (ICMT), 2011 International Conference on. pp. 1892–1896.

Loew, M. H., 2000. Feature extraction. Handbook of medical imaging 2, 273–342.

Loizou, C., Christodoulou, C., Pattichis, C., Istepanian, R., Pantziaris, M., Nicolaides, A., 2002. Speckle reduction in ultrasound images of atherosclerotic carotid plaque.

In: DSP 14th International Conference on Digital Signal Processing. Vol. 2. IEEE, pp. 525–528.

Loizou, C. P., Pattichis, C. S., 2008. Despeckle filtering algorithms and software for ultrasound imaging. Synthesis lectures on algorithms and software in engineering 1 (1), 1–166.

Loizou, C. P., Pattichis, C. S., Christodoulou, C. I., Istepanian, R. S. H., Pantziaris, M., Nicolaides, A., Oct. 2005. Comparative evaluation of despeckle filtering in ultrasound imaging of the carotid artery. IEEE transactions on ultrasonics, ferroelectrics, and frequency control 52 (10), 1653–69.

Loizou, C. P., Pattichis, C. S., Pantziaris, M., Tyllis, T., Nicolaides, A., 2006. Quality evaluation of ultrasound imaging in the carotid artery based on normalization and speckle reduction filtering. Medical and Biological Engineering and Computing 44 (5), 414–426.

Loizou, C. P., Theofanous, C., Pantziaris, M., Kasparis, T., 2014a. Despeckle filtering software toolbox for ultrasound imaging of the common carotid artery. Computer methods and programs in biomedicine 114 (1), 109–124.

Loizou, C. P., Theofanous, C., Pantziaris, M., Kasparis, T., 2014b. Despeckle filtering software toolbox for ultrasound imaging of the common carotid artery. Computer methods and programs in biomedicine 114 (1), 109–124.

Long, P., Cat, P. T., Dec 2009. Real-time speckle reducing by cellular neural network. In: Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on. pp. 1–4.

Loupas, T., McDicken, W., Allan, P., Jan 1989. An adaptive weighted median filter for speckle suppression in medical ultrasonic images. IEEE Transactions on Circuits and Systems 36 (1), 129–135.

Lu, H., Setiono, R., Liu, H., 1996. Effective data mining using neural networks. IEEE Transactions on Knowledge and Data Engineering 8 (6), 957–961.

Lu, L., Zhou, Y., Panetta, K., Agaian, S., 2010. Comparative study of histogram equalization algorithms for image enhancement. In: SPIE Defense, Security, and Sensing. International Society for Optics and Photonics, pp. 770811–770811.

Luboz, V., Perrier, A., Stavness, I., Lloyd, J. E., Bucki, M., Cannard, F., Diot, B., Vuillerme, N., Payan, Y., 2014. Foot ulcer prevention using biomechanical modelling. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization 2 (4), 189–196.

Ma, J., Perkins, S., 2003. Time-series novelty detection using one-class support vector machines. In: Neural Networks, 2003. Proceedings of the International Joint Conference on. Vol. 3. IEEE, pp. 1741–1745.

Malathi, G., Shanthi, V., Dec 2009. Wavelet based features for ultrasound placenta images classification. In: 2nd International Conference on Emerging Trends in Engineering and Technology (ICETET). pp. 341–345.

Mallat, S. G., 1989. A theory for multiresolution signal decomposition: the wavelet representation. IEEE transactions on pattern analysis and machine intelligence 11 (7), 674–693.

Măndoiu, I., Zelikovsky, A., 2007. Bioinformatics Research and Applications: Third International Symposium,ISBRA 2007, Atlanta, GA, USA, May 7-10, 2007, Proceedings. Lecture Notes in Computer Science. Springer Berlin Heidelberg.

Mangin, J.-F., Frouin, V., Bloch, I., Régis, J., López-Krahe, J., 1995. From 3d magnetic resonance images to structural representations of the cortex topography using topology preserving deformations. Journal of Mathematical Imaging and Vision 5 (4), 297–318.

Manian, V., Vasquez, R., Katiyar, P., Oct 2000. Texture classification using logical operators. IEEE Transactions on Image Processing 9 (10), 1693–1703.

Markou, M., Singh, S., 2006. A neural network-based novelty detector for image sequence analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 28 (10), 1664–1677.

Martinez, D., 1998. Neural tree density estimation for novelty detection. IEEE Transactions on Neural Networks 9 (2), 330–338.

Martínez-Trinidad, J., Ochoa, J., Kittler, J., for Pattern Recognition, I. A., 2006. Progress in Pattern Recognition, Image Analysis and Applications: 11th Iberoamerican Congress on Pattern Recognition, CIARP 2006, Cancún, Mexico, November 14-17, 2006, Proceedings. Image Processing, Computer Vision, Pattern Recognition, and Graphics. Springer.

Matthews, B. W., 1975. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. Biochimica et Biophysica Acta (BBA)-Protein Structure 405 (2), 442–451.

McLachlan, G. J., Basford, K. E., 1988. Mixture models: Inference and applications to clustering. Vol. 84. Marcel Dekker.

McNally, E. G., Shetty, S., 2010. Plantar fascia: imaging diagnosis and guided treatment. In: Seminars in musculoskeletal radiology. Vol. 14. p. 334.

McPoil, T. G., Martin, R. L., Cornwall, M. W., Wukich, D. K., Irrgang, J. J., Godges, J. J., 2008. Heel pain - plantar fasciitis: clinical practice guidelines linked to the international classification of function, disability, and health from then orthopaedics section of the american physical therapy association. Journal of Orthopaedic & Sports Physical Therapy 38 (4), 648.

Medeiros, F., Mascarenhas, N. D., Marques, R. C., Laprano, C. M., 2002. Edge preserving wavelet speckle filtering. In: Image Analysis and Interpretation, 2002. Proceedings. Fifth IEEE Southwest Symposium on. IEEE, pp. 281–285.

Megalooikonomou, V., Zhang, J., Kontos, D., Bakic, P. R., 2007. Analysis of texture patterns in medical images with an application to breast imaging. In: Medical Imaging. International Society for Optics and Photonics, pp. 651421–651421.

Metz, C. E., 1978. Basic principles of roc analysis. In: Seminars in nuclear medicine. Vol. 8. Elsevier, pp. 283–298.

Michailovich, O., Tannenbaum, A., Jan 2006. Despeckling of medical ultrasound images. IEEE Transactions on Ultrasonics, Ferroelectrics, and Frequency Control 53 (1), 64–78.

Milkowski, A., Li, Y., Becker, D., Ishrak, S. O., 2009. Speckle reduction imaging. Technical White Paper–General Electric Health Care (Ultrasound). Last accessed on July 9, 26.

Miller, L. E., Latt, D. L., 2015. Chronic plantar fasciitis is mediated by local hemodynamics: implications for emerging therapies. North American journal of medical sciences 7 (1), 1.

Moody, J., Darken, C. J., 1989. Fast learning in networks of locally-tuned processing units. Neural computation 1 (2), 281–294.

Moya, M. M., Hush, D. R., 1996. Network constraints and multi-objective optimization for one-class classification. Neural Networks 9 (3), 463–474.

Moya, M. M., Koch, M. W., Hostetler, L. D., 1993. One-class classifier networks for target recognition applications. Tech. rep., Sandia National Labs., Albuquerque, NM (United States).

Murthy, S. K., Kasif, S., Salzberg, S., 1994. A system for induction of oblique decision trees. Journal of artificial intelligence research 2, 1–32.

Murthy, S. K., Kasif, S., Salzberg, S., Beigel, R., 1993. Oc1: A randomized algorithm for building oblique decision trees. In: Proceedings of AAAI. Vol. 93. pp. 322–327.

Nabney, I., 2002. NETLAB: algorithms for pattern recognition. Springer Science & Business Media.

Nason, G., 1995. Choice of the threshold parameter in wavelet function estimation. In: Wavelets and statistics. Springer, pp. 261–280.

National Electrical Manufacturers Association (NEMA), 2011. Digital imaging and communications in medicine (DICOM). Part 14: Grayscale Standard Display Function. Rosslyn, VI: NEMA. Tech. rep.

Neufeld, S. K., Cerrato, R., 2008. Plantar fasciitis: evaluation and treatment. JAAOS-Journal of the American Academy of Orthopaedic Surgeons 16 (6), 338–346.

Nieminen, A., Heinonen, P., Neuvo, Y., 1987. A new class of detail-preserving filters for image processing. IEEE Transactions on Pattern Analysis and Machine Intelligence (1), 74–90.

Niranjan, M., Fallside, F., 1990. Neural networks and radial basis functions in classifying static speech patterns. Computer Speech & Language 4 (3), 275–289.

Noble, J. a., Boukerroui, D., 2006. Ultrasound image segmentation: A survey. IEEE Transactions on Medical Imaging 25 (8), 987–1010.

Ojala, T., Pietikäinen, M., Harwood, D., 1996. A comparative study of texture measures with classification based on featured distributions. Pattern recognition 29 (1), 51–59.

Om, H., Biswas, M., 2012. An improved image denoising method based on wavelet thresholding.

Orr, M. J., et al., 1996. Introduction to radial basis function networks.

Osuna, E., Freund, R., Girosi, F., 1997. Support vector machines: Training and applications.

Osuna, E., Girosi, F., 1998. Reducing the run-time complexity of support vector machines. In: International Conference on Pattern Recognition (submitted).

Othman, M. F., Basri, M. A. M., 2011. Probabilistic neural network for brain tumor classification. In: Second International Conference on Intelligent Systems, Modelling and Simulation (ISMS). IEEE, pp. 136–138.

Otsu, N., 1975. A threshold selection method from gray-level histograms. Automatica 11 (285-296), 23–27.

Ozkan, M., Dawant, B. M., Maciunas, R. J., 1993. Neural-network-based segmentation of multi-modal medical images: a comparative and prospective study. IEEE transactions on Medical Imaging 12 (3), 534–544.

Pal, N. R., Pal, S. K., 1993. A review on image segmentation techniques. Pattern Recognition 26 (9), 1277 – 1294.

Park, C., Huang, J. Z., Ding, Y., 2010. A computable plug-in estimator of minimum volume sets for novelty detection. Operations Research 58 (5), 1469–1480.

Park, J. W., Yoon, K., Chun, K. S., Lee, J. Y., Park, H. J., Lee, S. Y., Lee, Y. T., 2014. Long-term outcome of low-energy extracorporeal shock wave therapy for plantar fasciitis: Comparative analysis according to ultrasonographic findings. Annals of Rehabilitation Medicine 38 (4), 534–540.

Parzen, E., 1962. On estimation of a probability density function and mode. The annals of mathematical statistics 33 (3), 1065–1076.

Pathak, S., Grimm, P., Chalana, V., Kim, Y., Oct 1998. Pubic arch detection in transrectal ultrasound guided prostate cancer therapy. IEEE transactions on medical imaging 17 (5), 762–771.

Pathak, S. D., Chalana, V., Kim, Y., 1997. Interactive automatic fetal head measurements from ultrasound images using multimedia computer technology. Ultrasound in Medicine & Biology 23 (5), 665 – 673.

Peck, C. C., Dhawan, A. P., 1993. A review and critique of genetic algorithm theories. J of Evolutionary Computing, 39–80.

Perona, P., Malik, J., Jul 1990. Scale-space and edge detection using anisotropic diffusion. Pattern Analysis and Machine Intelligence, IEEE Transactions on 12 (7), 629–639.

Perry, N., Broeders, M., De Wolf, C., Törnberg, S., Holland, R., von Karsa, L., Puthaar, E., 2006. European guidelines for quality assurance in breast cancer screening and diagnosis fourth edition. Luxembourg: Office for Official Publications of the European Communities.

Pfeffer, G., Bacchetti, P., Deland, J., Lewis, A., Anderson, R., Davis, W., Alvarez, R., Brodsky, J., Cooper, P., Frey, C., et al., 1999. Comparison of custom and prefabricated orthoses in the initial treatment of proximal plantar fasciitis. Foot & Ankle International 20 (4), 214–221.

Pham, D. L., Xu, C., Prince, J. L., 1998. A survey of current methods in medical image segmentation.

Pimentel, M. A., Clifton, D. A., Clifton, L., Tarassenko, L., 2014. A review of novelty detection. Signal Processing 99, 215–249.
URL http://www.robots.ox.ac.uk/~davidc/publications_NDtool.php

Ping, L., 2004. A survey on threshold selection of image segmentation. Journal of Image and Graphics, 8692.

Pitas, I., Venetsanopoulos, A. N., 1990. Nonlinear digital filters. Vol. 84. Springer Science & Business Media.

Pizer, S. M., 2003. The medical image display and analysis group at the university of north carolina: Reminiscences and philosophy. IEEE Transactions on Medical Imaging 22 (1), 2–10.

Pizer, S. M., Amburn, E. P., Austin, J. D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar Romeny, B., Zimmerman, J. B., Zuiderveld, K., 1987. Adaptive histogram equalization and its variations. Computer vision, graphics, and image processing 39 (3), 355–368.

Pizurica, A., Philips, W., Lemahieu, I., Acheroy, M., 2003a. A versatile wavelet domain noise filtration technique for medical imaging. IEEE transactions on medical imaging 22 (3), 323–331.

Pizurica, A., Philips, W., Lemahieu, I., Acheroy, M., March 2003b. A versatile wavelet domain noise filtration technique for medical imaging. IEEE Transactions on Medical Imaging 22 (3), 323–331.

Platt, J., 1998. Sequential minimal optimization: A fast algorithm for training support vector machines.

Platt, J., et al., 1999. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. Advances in large margin classifiers 10 (3), 61–74.

Pope, J., 1999. Medical Physics: Imaging. Heinemann advanced science. Pearson Education.

Portilla, J., Strela, V., Wainwright, M., Simoncelli, E., Nov 2003. Image denoising using scale mixtures of gaussians in the wavelet domain. IEEE Transactions on Image Processing 12 (11), 1338–1351.

Prakash, R. M., Kumari, R. S. S., 2017. Spatial fuzzy c means and expectation maximization algorithms with bias correction for segmentation of mr brain images. Journal of Medical Systems 41 (1), 15.

Prewitt, J. M., 1970. Object enhancement and extraction. Picture processing and Psychopictorics 10 (1), 15–19.

Priddy, K., Keller, P., 2005. Artificial Neural Networks: An Introduction. SPIE tutorial texts. Society of Photo Optical.

Prinosil, J., Smekal, Z., Bartusek, K., March 2010. Wavelet thresholding techniques in mri domain. In: 2010 International Conference on Biosciences (BIOSCIENCESWORLD),. pp. 58–63.

Puttaswamaiah, R., Chandran, P., 2007. Degenerative plantar fasciitis: A review of current concepts. The Foot 17 (1), 3 – 9.

Quinlan, J. R., 1993. C4.5: Programs for machine learning morgan kaufmann san mateo. Morgan Kauffmann 38.

Quinn, J., Williams, C., 2007. Known unknowns: Novelty detection in condition monitoring. Pattern Recognition and Image Analysis, 1–6.

Rabbani, H., Vafadust, M., Abolmaesumi, P., Gazor, S., Sept 2008. Speckle noise reduction of medical ultrasound images in complex wavelet domain using mixture priors. IEEE Transactions on Biomedical Engineering 55 (9), 2152–2160.

Rajan, J., Kannan, K., Kaimal, M., 2008. An improved hybrid model for molecular image denoising. Journal of Mathematical Imaging and Vision 31 (1), 73–79.

Rajini, N., Bhavani, R., June 2011. Classification of mri brain images using k-nearest neighbor and artificial neural network. In: Recent Trends in Information Technology (ICRTIT), 2011 International Conference on. pp. 563–568.

Rajpoot, N. M., Wilson, R. G., Meyer, F. G., Coifman, R. R., 2003. Adaptive wavelet packet basis selection for zerotree image coding. IEEE Transactions on Image Processing 12 (12), 1460–1472.

Rao, K. R., Yip, P., 2014. Discrete cosine transform: algorithms, advantages, applications. Academic press.

Ratsch, G., Mika, S., Scholkopf, B., Muller, K.-R., 2002. Constructing boosting algorithms from svms: an application to one-class classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 24 (9), 1184–1199.

Remus, J. J., Morton, K. D., Torrione, P., Tantum, S. L., Collins, L. M., et al., 2008. Comparison of a distance-based likelihood ratio test and k-nearest neighbor classification methods. In: IEEE Workshop on Machine Learning for Signal Processing. MLSP 2008. IEEE, pp. 362–367.

Reuter, M., Biasotti, S., Giorgi, D., Patanè, G., Spagnuolo, M., 2009. Discrete laplace-beltrami operators for shape analysis and segmentation. Computers & Graphics 33 (3), 381–390.

Ritenour, E., Nelson, T., Raff, U., 1984. Applications of the median filter to digital radiographic images. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'84. Vol. 9. IEEE, pp. 251–254.

Ritter, G., Gallegos, M. T., 1997. Outliers in statistical pattern recognition and an application to automatic chromosome classification. Pattern Recognition Letters 18 (6), 525–539.

Rizi, F. Y., Noubari, H. A., Setarehdan, S. K., 2011. Biomedical image and signal denoising using dual tree complex wavelet transform. Vol. 8285. pp. 828547–828547–6.

Roffo, G., Melzi, S., Castellani, U., Vinciarelli, A., Oct 2017. Infinite latent feature selection: A probabilistic latent graph-based ranking approach. In: 2017 IEEE International Conference on Computer Vision (ICCV).

Roffo, G., Melzi, S., Cristani, M., 2015a. Infinite feature selection. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4202–4210.

Roffo, G., Melzi, S., Cristani, M., 2015b. Infinite feature selection. 2015 IEEE International Conference on Computer Vision (ICCV), pp. 4202–4210.

Rosa, R., Monteiro, F., 2014. Performance analysis of speckle ultrasound image filtering. Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization 0 (0), 1–9.

Rubinstein, R. Y., 1997. Optimization of computer simulation models with rare events. European Journal of Operational Research 99 (1), 89–112.

Rueda, S., Fathima, S., Knight, C. L., Yaqub, M., Papageorghiou, A. T., Rahmatullah, B., Foi, A., Maggioni, M., Pepe, A., Tohka, J., Stebbing, R. V., McManigle, J. E., Ciurte, A., Bresson, X., Cuadra, M. B., Sun, C., Ponomarev, G. V., Gelfand, M. S., Kazanov, M. D., Wang, C.-W., Chen, H.-C., Peng, C.-W., Hung, C.-M., Noble, J. A., Apr. 2014. Evaluation and comparison of current fetal ultrasound image segmentation methods for biometric measurements: a grand challenge. IEEE transactions on medical imaging 33 (4), 797–813.

Rumelhart, D. E., Hinton, G. E., Williams, R. J., 1988. Learning representations by back-propagating errors. Cognitive modeling 5 (3), 1.

Russ, J., 2016. The Image Processing Handbook, Seventh Edition. CRC Press.

Saber, N., Diab, H., Nassar, W., Razaak, H. A., 2012. Ultrasound guided local steroid injection versus extracorporeal shockwave therapy in the treatment of plantar fasciitis. Alexandria Journal of Medicine 48 (1), 35–42.

Sahoo, P. K., Soltani, S., Wong, A. K., 1988. A survey of thresholding techniques. Computer vision, graphics, and image processing 41 (2), 233–260.

Saraniya, O., Ezhilarasi, M., 2014. Despeckling of medical ultrasound images by wavelet threshold optimisation. International Journal of Biomedical Engineering and Technology 15 (1), 1–17, pMID: 60985.

Sarwal, A., Dhawan, A. P., 1998. Segmentation of coronary arteries using radial basis function neural-network. CIT. Journal of computing and information technology 6 (2), 135–148.

Saurabh, A., Kumar, A., Anitha, U., March 2015. Performance analysis of various wavelet thresholding techniques for despeckiling of sonar images. In: Signal Processing, Communication and Networking (ICSCN), 2015 3rd International Conference on. pp. 1–7.

Schölkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., Platt, J. C., et al., 1999. Support vector method for novelty detection. In: NIPS. Vol. 12. pp. 582–588.

Sebbahi, A., Herment, A., de Cesare, A., Mousseaux, E., 1997. Multimodality cardiovascular image segmentation using a deformable contour model. Computerized Medical Imaging and Graphics 21 (2), 79–89.

Selesnick, I., Baraniuk, R., Kingsbury, N., Nov 2005. The dual-tree complex wavelet transform. IEEE Signal Processing Magazine 22 (6), 123–151.

Selesnick, I. W., 2001. Hilbert transform pairs of wavelet bases. Signal Processing Letters, IEEE 8 (6), 170–173.

Selesnick, I. W., 2002. The design of approximate hilbert transform pairs of wavelet bases. IEEE Transactions on Signal Processing 50 (5), 1144–1152.

Sendur, L., Selesnick, I., Nov 2002a. Bivariate shrinkage functions for wavelet-based denoising exploiting interscale dependency. IEEE Transactions on Signal Processing 50 (11), 2744–2756.

Sendur, L., Selesnick, I., Dec 2002b. Bivariate shrinkage with local variance estimation. IEEE Signal Processing Letters 9 (12), 438–441.

Serbes, G., Aydin, N., Nov 2010. Denoising performance of modified dual tree complex wavelet transform. In: 10th IEEE International Conference on Information Technology and Applications in Biomedicine (ITAB). pp. 1–4.

Shazia, A., Davinder, P., Singh, B., 2011. Plantar heel pain. Clinical Focus Primary Care 5, 128–33.

Shental, N., Bar-Hillel, A., Hertz, T., Weinshall, D., 2003. Computing gaussian mixture models with em using equivalence constraints. In: NIPS. Vol. 50. p. 112.

Shi, J., Malik, J., 2000. Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22 (8), 888–905.

Shi, X., Cheng, H., Hu, L., Ju, W., Tian, J., 2010. Detection and classification of masses in breast ultrasound images. Digital Signal Processing 20 (3), 824–836.

Shih, F., 2009. Image Processing and Mathematical Morphology: Fundamentals and Applications. CRC Press.

Shih, F., 2010. Image Processing and Pattern Recognition: Fundamentals and Techniques. Wiley.

Shui, P.-L., Oct 2005a. Image denoising algorithm via doubly local wiener filtering with directional windows in wavelet domain. IEEE Transaction on Signal Processing Letters 12 (10), 681–684.

Shui, P.-L., Oct 2005b. Image denoising algorithm via doubly local wiener filtering with directional windows in wavelet domain. IEEE, Signal Processing Letters 12 (10), 681–684.

Shui, P.-l., Zhao, Y.-B., 2007. Image denoising algorithm using doubly local wiener filtering with block-adaptive windows in wavelet domain. Signal Processing 87 (7), 1721–1734.

Silverman, B. W., 2018. Density estimation for statistics and data analysis. Routledge.

Singh, Y. K., Parui, S. K., Jan. 2006. Isitra: A generalized way of signal decomposition and reconstruction. Digital Signal Processing 16 (1), 3–23.

Sivakumar, R., Gayathri, M., Nedumaran, D., Dec 2010. Speckle filtering of ultrasound b-scan images - a comparative study between spatial and diffusion filters. In: IEEE Conference on Open Systems (ICOS). pp. 80–85.

Sokolova, M., Lapalme, G., 2009. A systematic analysis of performance measures for classification tasks. Information Processing and Management 45 (4), 427–437.

Solbo, S., Eltoft, T., April 2008. A stationary wavelet-domain wiener filter for correlated speckle. IEEE Transactions on Geoscience and Remote Sensing 46 (4), 1219–1230.

Sonka, M., Hlavac, V., Boyle, R., 2014. Image Processing, Analysis, and Machine Vision. Cengage Learning.

Sonka, M., Zhang, X., Siebes, M., Bissing, M., DeJong, S., Collins, S. M., McKay, C., Dec 1995. Segmentation of intravascular ultrasound images: a knowledge-based approach. IEEE Transactions on Medical Imaging 14 (4), 719–732.

Sridevi, S., Sundaresan, M., 2013. Survey of image segmentation algorithms on ultrasound medical images. In: 2013 International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME). IEEE, pp. 215–220.

Srivastava, R., Gupta, J., Parthasarthy, H., 2010. Comparison of pde based and other techniques for speckle reduction from digitally reconstructed holographic images. Optics and Lasers in Engineering 48 (5), 626–635.

Surace, C., Worden, K., 2010. Novelty detection in a changing environment: a negative selection approach. Mechanical Systems and Signal Processing 24 (4), 1114–1128.

Suri, J., 2008. Advances in Diagnostic and Therapeutic Ultrasound Imaging. Artech House bioinformatics & biomedical imaging series. Artech House.

Suykens, J. A., 2017. Efficient sparse approximation of support vector machines solving a kernel lasso. In: Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications: 21st Iberoamerican Congress, CIARP 2016, Lima, Peru, November 8–11, 2016, Proceedings. Vol. 10125. Springer, p. 208.

Szabo, T., 2013. Diagnostic Ultrasound Imaging: Inside Out. Biomedical Engineering. Elsevier Science.

Tadeusiewicz, R., Ogiela, M. R., 2006. Automatic image understanding a new paradigm for intelligent medical image analysis. Bio-Algorithms and Med-Systems 2 (3), 3–9.

Taher, F., Sammouda, R., 2011. Lung cancer detection by using artificial neural network and fuzzy clustering methods. In: GCC Conference and Exhibition (GCC), 2011 IEEE. IEEE, pp. 295–298.

Tan, P.-N., Steinbach, M., Kumar, V., 2005. Chapter 6. association analysis: basic concepts and algorithms. introduction to data mining.

Tarassenko, L., Hayton, P., Cerneaz, N., Brady, M., 1995. Novelty detection for the identification of masses in mammograms.

Tax, D. M., Duin, R. P., 1999a. Data domain description using support vectors. In: ESANN. Vol. 99. pp. 251–256.

Tax, D. M., Duin, R. P., 1999b. Support vector domain description. Pattern recognition letters 20 (11), 1191–1199.

Telea, A., 2014. Data Visualization: Principles and Practice, Second Edition. Taylor & Francis.

The Royal College of Radiologists, 2014. Picture archiving and communication systems (PACS) and guidelines on diagnostic display devices. London: RCR. Tech. rep.

Theodoridis, S., Koutroumbas, K., 2006. Pattern Recognition. Pattern Recognition Series. Elsevier Science.

Toennies, K., 2012. Guide to Medical Image Analysis: Methods and Algorithms. Advances in Computer Vision and Pattern Recognition. Springer London.

Torkamani-Azar, F., Tait, K., Nov 1996. Image recovery using the anisotropic diffusion equation. IEEE Transactions on Image Processing 5 (11), 1573–1578.

Tsai, D.-M., 1995. A fast thresholding selection procedure for multimodal and unimodal histograms. Pattern Recognition Letters 16 (6), 653–666.

Tuceryan, M., Jain, A. K., 1993. Handbook of pattern recognition &amp; computer vision. World Scientific Publishing Co., Inc., River Edge, NJ, USA, Ch. Texture Analysis, pp. 235–276.

Tuong Vinh, H., ROUAT, J., 2001. A novelty detector using a network of integrate-and-fire neurons. ECDL Module 4: Spreadsheets: ECDL-the European PC standard 4, 103.

Uddin, M., Tahtali, M., Lambert, A., Pickering, M., 2013. Speckle reduction for ultrasound images using nonlinear multi-scale complex wavelet diffusion. In: IEEE International Conference on Signal and Image Processing Applications (ICSIPA). pp. 31–36.

Udupa, J. K., Leblanc, V. R., Zhuge, Y., Imielinska, C., Schmidt, H., Currie, L. M., Hirsch, B. E., Woodburn, J., 2006. A framework for evaluating image segmentation algorithms. Computerized Medical Imaging and Graphics 30 (2), 75–87.

Udupa, J. K., Samarasekera, S., 1996. Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. Graphical models and image processing 58 (3), 246–261.

Umbaugh, S., 2005. Computer Imaging: Digital Image Analysis and Processing. A CRC Press book. Taylor & Francis.

Umbaugh, S., 2010. Digital Image Processing and Analysis: Human and Computer Vision Applications with CVIPtools, Second Edition. Taylor & Francis.

Unger, H., Meesad, P., Boonkrong, S., 2015. Recent Advances in Information and Communication Technology 2015. Advances in Intelligent Systems and Computing. Springer International Publishing.

Vapnik, V., 1998. The support vector method of function estimation. In: Nonlinear Modeling. Springer, pp. 55–85.

Vapnik, V., 2013. The nature of statistical learning theory. Springer science & business media.

Vidakovic, B., 2009. Statistical modeling by wavelets. Vol. 503. John Wiley & Sons.

Wagner, R., Smith, S., Sandrik, J., Lopez, H., May 1983. Statistics of speckle in ultrasound b-scans. IEEE Transactions on Sonics and Ultrasonics 30 (3), 156–163.

Wang, H., 2002. Nearest neighbours without k: A classification formalism based on probability. Technical Report, Faculty of Informatics.

Wang, H.-Z., He, X.-H., Zai, W.-j., 2007. Texture image retrieval using dual-tree complex wavelet transform. In: IEEE International Conference on Wavelet Analysis and Pattern Recognition, ICWAPR'07. Vol. 1. IEEE, pp. 230–234.

Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E., April 2004. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing 13 (4).

Wearing, S. C., Smeathers, J. E., Sullivan, P. M., Yates, B., Urry, S. R., Dubois, P., 2007. Plantar fasciitis: are pain and fascial thickness associated with arch shape and loading? Physical therapy 87 (8), 1002–1008.

Wearing, S. C., Smeathers, J. E., Yates, B., Sullivan, P. M., Urry, S. R., Dubois, P., 2004. Sagittal movement of the medial longitudinal arch is unchanged in plantar fasciitis. Medicine and science in sports and exercise 36 (10), 1761–1767.

Weickert, J., 1998. Anisotropic diffusion in image processing. Vol. 1. Teubner Stuttgart.

Wen, X., Yuan, H., Yang, C., Song, C., Duan, B., Zhao, H., Sept 2007a. Improved haar wavelet feature extraction approaches for vehicle detection. In: IEEE Intelligent Transportation Systems Conference. ITSC 2007. pp. 1050–1053.

Wen, X., Yuan, H., Yang, C., Song, C., Duan, B., Zhao, H., 2007b. Improved haar wavelet feature extraction approaches for vehicle detection. In: Intelligent Transportation Systems Conference, 2007. ITSC 2007. IEEE. IEEE, pp. 1050–1053.

Weszka, J. S., Dyer, C. R., Rosenfeld, A., 1976a. A comparative study of texture measures for terrain classification. IEEE Transactions on Systems, Man and Cybernetics SMC-6 (4), 269–285.

Weszka, J. S., Dyer, C. R., Rosenfeld, A., 1976b. A comparative study of texture measures for terrain classification. IEEE Transactions on Systems, Man, and Cybernetics (4), 269–285.

Whatmough, R., 1991. Automatic threshold selection from a histogram using the exponential hull. CVGIP: Graphical Models and Image Processing 53 (6), 592–600.

Williams, D., Hinton, G., 1986. Learning representations by back-propagating errors. Nature 323 (6088), 533–538.

Wu, C.-M., Chen, Y.-C., 1992. Statistical feature matrix for texture analysis. CVGIP: Graphical Models and Image Processing 54 (5), 407–419.

Wu, C.-M., Chen, Y.-C., Hsieh, K.-S., 1992. Texture features for classification of ultrasonic liver images. IEEE Transactions on medical imaging 11 (2), 141–152.

Wu, K.-L., Yang, M.-S., 2002. Alternative c-means clustering algorithms. Pattern recognition 35 (10), 2267–2278.

Wu, P., Wang, Z., June 2010. The image edge detection algorithm based on wavelet denoising and mathematics morphology. In: Systems and Control in Aeronautics and Astronautics (ISSCAA), 2010 3rd International Symposium on. pp. 990–995.

Wu, S., Zhu, Q., Xie, Y., July 2013. Evaluation of various speckle reduction filters on medical ultrasound images. In: Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE. pp. 1148–1151.

Yan, F., Zhang, H., Kube, C. R., 2005. A multistage adaptive thresholding method. Pattern recognition letters 26 (8), 1183–1191.

Yang, M.-S., Hu, Y.-J., Lin, K. C.-R., Lin, C. C.-L., 2002. Segmentation techniques for tissue differentiation in mri of ophthalmology using fuzzy clustering algorithms. Magnetic Resonance Imaging 20 (2), 173–179.

Yang, Y., Liu, X., 1999. A re-examination of text categorization methods. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval. ACM, pp. 42–49.

Yang, Y., Pedersen, J. O., 1997. A comparative study on feature selection in text categorization. In: Icml. Vol. 97. pp. 412–420.

Yezzi, A., J., Kichenassamy, S., Kumar, A., Olver, P., Tannenbaum, A., April 1997. A geometric snake model for segmentation of medical imagery. IEEE transactions on medical imaging 16 (2), 199–209.

Yoo, B., Nishimura, T., July 2009. A study of ultrasound images enhancement using adaptive speckle reducing anisotropic diffusion. In: IEEE International Symposium on Industrial Electronics, ISIE 2009. pp. 581–585.

You, Y. L., Kaveh, M., Oct 2000. Fourth-order partial differential equations for noise removal. IEEE Transactions on Image Processing 9 (10), 1723–1730.

Yu, J., Wang, Y., 2007. Molecular image segmentation based on improved fuzzy clustering. International journal of biomedical imaging 2007.

Yu, L., Han, Y., Berens, M. E., 2012. Stable gene selection from microarray data via sample weighting. IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB) 9 (1), 262–272.

Yu, L., Liu, H., 2004. Efficient feature selection via analysis of relevance and redundancy. The Journal of Machine Learning Research 5, 1205–1224.

Yu, Y., Acton, S., Nov 2002. Speckle reducing anisotropic diffusion. IEEE Transactions on Image Processing 11 (11), 1260–1270.

Yuan, W., Liu, J., Zhou, H.-B., Aug 2004. An improved knn method and its application to tumor diagnosis. In: Machine Learning and Cybernetics, 2004. Proceedings of 2004 International Conference on. Vol. 5. pp. 2836–2841 vol.5.

Yue, Y., Croitoru, M., Bidani, A., Zwischenberger, J., Clark, J. W., March 2006a. Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images. IEEE Transactions on Medical Imaging 25 (3), 297–311.

Yue, Y., Croitoru, M., Bidani, A., Zwischenberger, J., Clark, J. W., March 2006b. Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images. Medical Imaging, IEEE Transactions on 25 (3), 297–311.

Yue, Y., Croitoru, M. M., Bidani, A., Zwischenberger, J. B., Clark, J. W., 2006c. Ultrasound speckle suppression and edge enhancement using multiscale nonlinear wavelet diffusion. In: Engineering in Medicine and Biology Society, 2005. IEEE-EMBS 2005. 27th Annual International Conference of the. IEEE, pp. 6429–6432.

Yun, J., Zhanhuai, L., Yong, W., Longbo, Z., 2006. A better classifier based on rough set and neural network for medical images. In: Sixth IEEE International Conference on Data Mining Workshops. ICDM Workshops. IEEE, pp. 853–857.

Zadeh, L. A., 1965. Fuzzy sets. Information and control 8 (3), 338–353.

Zadeh, L. A., Fu, K.-S., Tanaka, K., 2014. Fuzzy sets and their applications to cognitive and decision processes: Proceedings of the us–japan seminar on fuzzy sets and their applications, held at the university of california, berkeley, california, july 1-4, 1974. Academic press.

Zaffalon, M., Hutter, M., 2002. Robust feature selection using distributions of mutual information. In: Proceedings of the 18th International Conference on Uncertainty in Artificial Intelligence (UAI-2002). pp. 577–584.

Zang, H., Wang, Z., Zheng, Y., Aug 2009. Analysis of signal de-noising method based on an improved wavelet thresholding. In: ICEMI 9th International Conference on Electronic Measurement Instruments. pp. 1–987–1–990.

Zhai, J.-H., Wang, X.-Z., Zhang, S.-F., 2007. Rough-neural image classification using wavelet transform. In: IEEE International Conference on Machine Learning and Cybernetics. Vol. 6. IEEE, pp. 3045–3050.

Zhang, D., Wong, A., Indrawan, M., Lu, G., 2000. Content-based image retrieval using gabor texture features. IEEE Transactions PAMI, 13–15.

Zhang, G., Wu, J., Cui, Z., Dec 2008. Application of wavelet thresholding de-noising in dsa. In: Information Science and Engineering, 2008. ISISE '08. International Symposium on. Vol. 1. pp. 130–134.

Zhang, J., Wang, C., Cheng, Y., 2015. Comparison of despeckle filters for breast ultrasound images. Circuits, Systems, and Signal Processing 34 (1), 185–208.

Zhang, Y., 2006. Advances in Image and Video Segmentation. IGI Global research collection. IGI Global.

Zhao, Y., Georganas, N. D., Petriu, E., May 2010. Applying contrast-limited adaptive histogram equalization and integral projection for facial feature enhancement and detection. In: Instrumentation and Measurement Technology Conference (I2MTC), 2010 IEEE. pp. 861–866.

Zhiyun, L., Tao, J., Zengwu, S., 2013. Meta-analysis of high-energy extracorporeal shock wave therapy in recalcitrant plantar fasciitis. Swiss Med Wkly 143, w13825.

Zhong, S., Cherkassky, V., 2000. Image denoising using wavelet thresholding and model selection. In: Image Processing, 2000. Proceedings. 2000 International Conference on. Vol. 3. IEEE, pp. 262–265.

Zimmer, Y., Tepper, R., Akselrod, S., 1996. A two-dimensional extension of minimum cross entropy thresholding for the segmentation of ultrasound images. Ultrasound in Medicine & Biology 22 (9), 1183–1190.

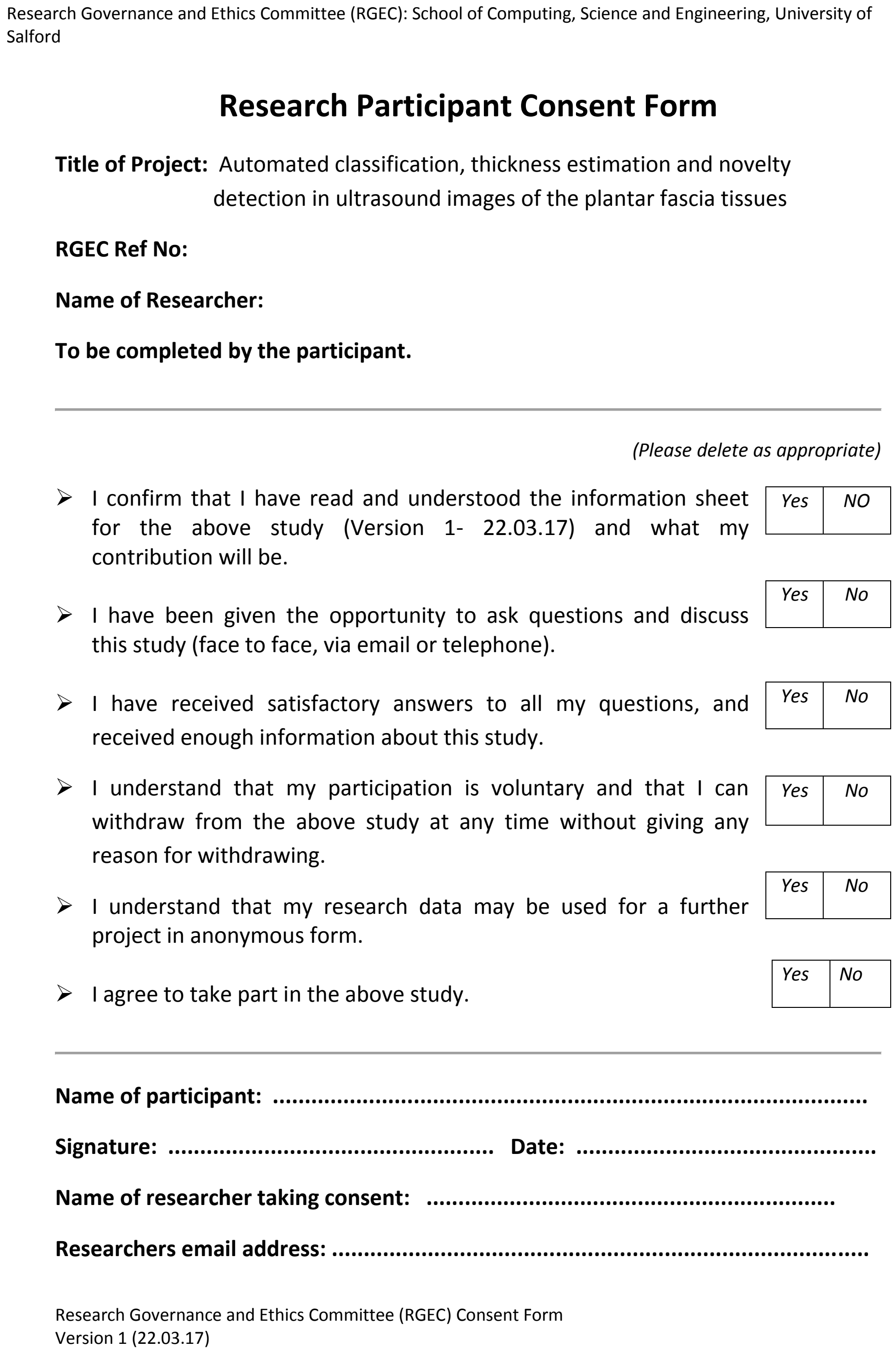
Zong, X., Laine, A., Geiser, E. A., Aug 1998. Speckle reduction and contrast enhancement of echocardiograms via multiscale nonlinear processing. IEEE Transactions on Medical Imaging 17 (4), 532–540.

Zuiderveld, K., 1994. Graphics gems iv. Academic Press Professional, Inc., San Diego, CA, USA, Ch. Contrast Limited Adaptive Histogram Equalization, pp. 474–485.

Zurada, J. M., 1992. Introduction to artificial neural systems. Vol. 8. West St. Paul.

Appendix A

Research Participant Consent Form and Participant Information Sheet



Research Governance and Ethics Committee (RGEC): School of Computing, Science and Engineering, University of Salford

Research Participant Consent Form

Title of Project: Automated classification, thickness estimation and novelty detection in ultrasound images of the plantar fascia tissues

RGEC Ref No:

Name of Researcher:

To be completed by the participant.

(Please delete as appropriate)

- I confirm that I have read and understood the information sheet for the above study (Version 1- 22.03.17) and what my contribution will be. *Yes* *NO*
- I have been given the opportunity to ask questions and discuss this study (face to face, via email or telephone). *Yes* *No*
- I have received satisfactory answers to all my questions, and received enough information about this study. *Yes* *No*
- I understand that my participation is voluntary and that I can withdraw from the above study at any time without giving any reason for withdrawing. *Yes* *No*
- I understand that my research data may be used for a further project in anonymous form. *Yes* *No*
- I agree to take part in the above study. *Yes* *No*

Name of participant: ....

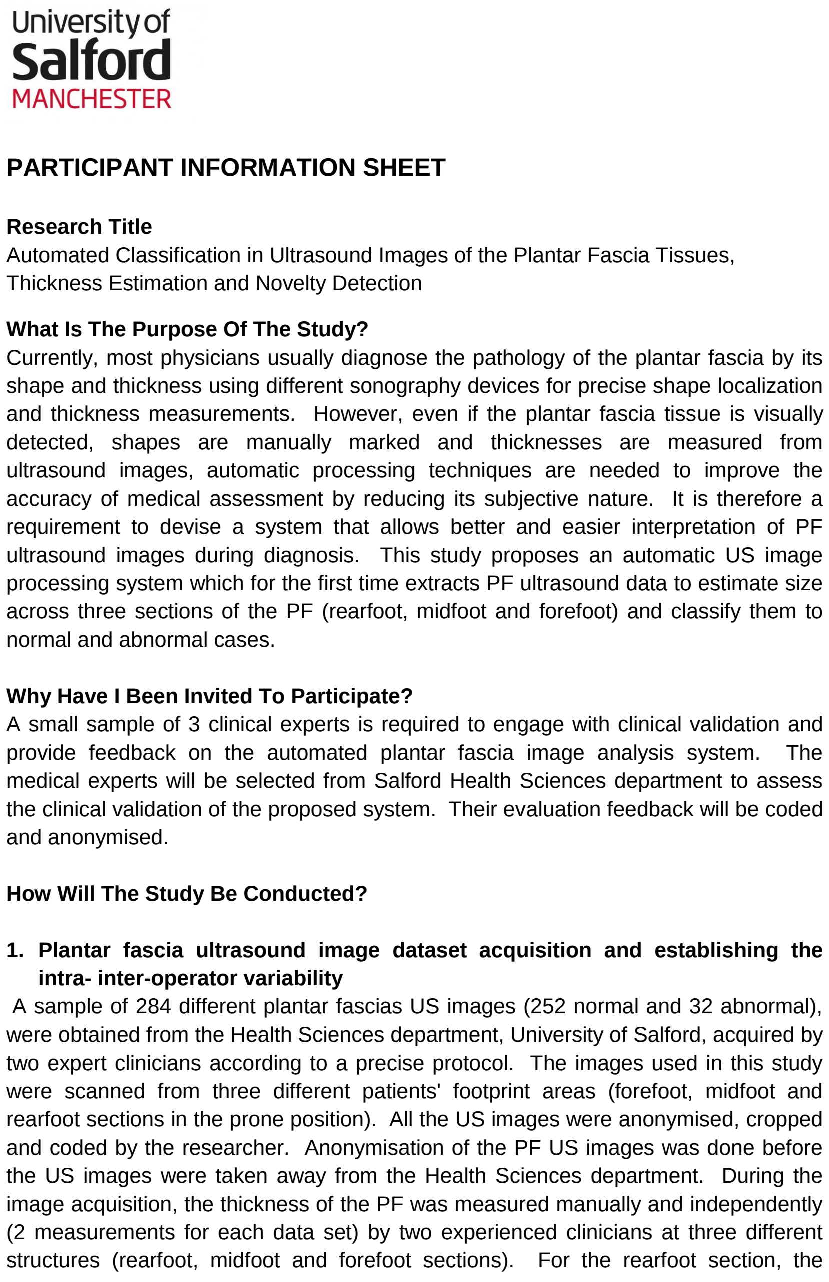
Signature: .... **Date:** ....

Name of researcher taking consent: ....

Researchers email address: ....

Research Governance and Ethics Committee (RGEC) Consent Form
Version 1 (22.03.17)

Figure A.1: Research Participant Consent Form



University of
Salford
MANCHESTER

PARTICIPANT INFORMATION SHEET

Research Title

Automated Classification in Ultrasound Images of the Plantar Fascia Tissues,
Thickness Estimation and Novelty Detection

What Is The Purpose Of The Study?

Currently, most physicians usually diagnose the pathology of the plantar fascia by its shape and thickness using different sonography devices for precise shape localization and thickness measurements. However, even if the plantar fascia tissue is visually detected, shapes are manually marked and thicknesses are measured from ultrasound images, automatic processing techniques are needed to improve the accuracy of medical assessment by reducing its subjective nature. It is therefore a requirement to devise a system that allows better and easier interpretation of PF ultrasound images during diagnosis. This study proposes an automatic US image processing system which for the first time extracts PF ultrasound data to estimate size across three sections of the PF (rearfoot, midfoot and forefoot) and classify them to normal and abnormal cases.

Why Have I Been Invited To Participate?

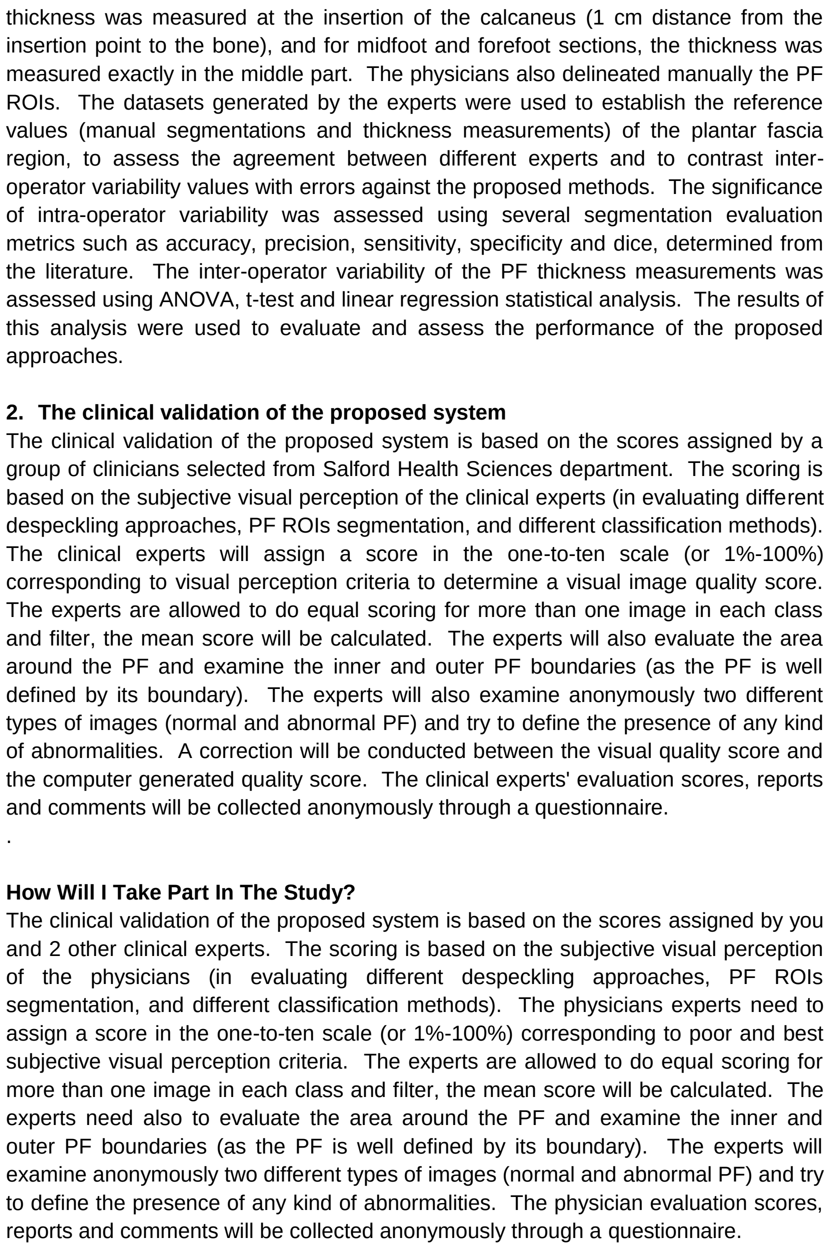
A small sample of 3 clinical experts is required to engage with clinical validation and provide feedback on the automated plantar fascia image analysis system. The medical experts will be selected from Salford Health Sciences department to assess the clinical validation of the proposed system. Their evaluation feedback will be coded and anonymised.

How Will The Study Be Conducted?

1. **Plantar fascia ultrasound image dataset acquisition and establishing the intra- inter-operator variability**

A sample of 284 different plantar fascias US images (252 normal and 32 abnormal), were obtained from the Health Sciences department, University of Salford, acquired by two expert clinicians according to a precise protocol. The images used in this study were scanned from three different patients' footprint areas (forefoot, midfoot and rearfoot sections in the prone position). All the US images were anonymised, cropped and coded by the researcher. Anonymisation of the PF US images was done before the US images were taken away from the Health Sciences department. During the image acquisition, the thickness of the PF was measured manually and independently (2 measurements for each data set) by two experienced clinicians at three different structures (rearfoot, midfoot and forefoot sections). For the rearfoot section, the

Figure A.2: Participant Information Sheet (page 1)



thickness was measured at the insertion of the calcaneus (1 cm distance from the insertion point to the bone), and for midfoot and forefoot sections, the thickness was measured exactly in the middle part. The physicians also delineated manually the PF ROIs. The datasets generated by the experts were used to establish the reference values (manual segmentations and thickness measurements) of the plantar fascia region, to assess the agreement between different experts and to contrast inter-operator variability values with errors against the proposed methods. The significance of intra-operator variability was assessed using several segmentation evaluation metrics such as accuracy, precision, sensitivity, specificity and dice, determined from the literature. The inter-operator variability of the PF thickness measurements was assessed using ANOVA, t-test and linear regression statistical analysis. The results of this analysis were used to evaluate and assess the performance of the proposed approaches.

2. The clinical validation of the proposed system

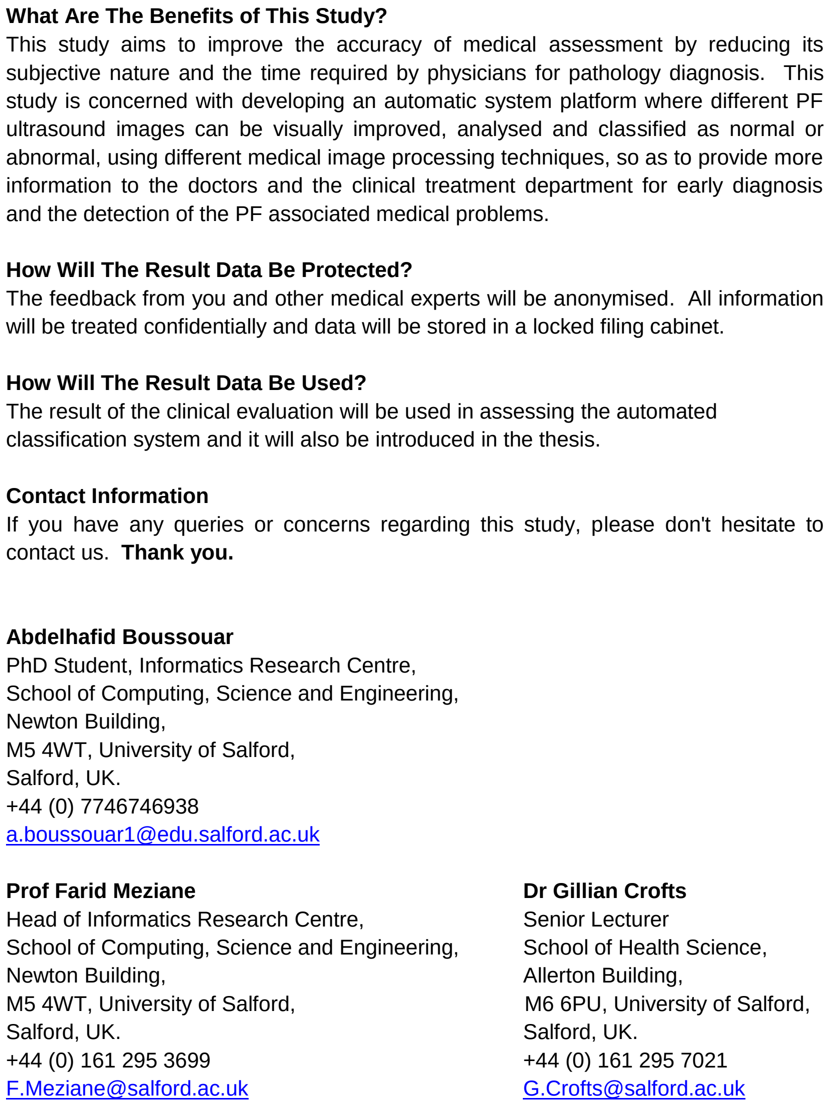
The clinical validation of the proposed system is based on the scores assigned by a group of clinicians selected from Salford Health Sciences department. The scoring is based on the subjective visual perception of the clinical experts (in evaluating different despeckling approaches, PF ROIs segmentation, and different classification methods). The clinical experts will assign a score in the one-to-ten scale (or 1%-100%) corresponding to visual perception criteria to determine a visual image quality score. The experts are allowed to do equal scoring for more than one image in each class and filter, the mean score will be calculated. The experts will also evaluate the area around the PF and examine the inner and outer PF boundaries (as the PF is well defined by its boundary). The experts will also examine anonymously two different types of images (normal and abnormal PF) and try to define the presence of any kind of abnormalities. A correction will be conducted between the visual quality score and the computer generated quality score. The clinical experts' evaluation scores, reports and comments will be collected anonymously through a questionnaire.

.

How Will I Take Part In The Study?

The clinical validation of the proposed system is based on the scores assigned by you and 2 other clinical experts. The scoring is based on the subjective visual perception of the physicians (in evaluating different despeckling approaches, PF ROIs segmentation, and different classification methods). The physicians experts need to assign a score in the one-to-ten scale (or 1%-100%) corresponding to poor and best subjective visual perception criteria. The experts are allowed to do equal scoring for more than one image in each class and filter, the mean score will be calculated. The experts need also to evaluate the area around the PF and examine the inner and outer PF boundaries (as the PF is well defined by its boundary). The experts will examine anonymously two different types of images (normal and abnormal PF) and try to define the presence of any kind of abnormalities. The physician evaluation scores, reports and comments will be collected anonymously through a questionnaire.

Figure A.3: Participant Information Sheet (page 2)



What Are The Benefits of This Study?

This study aims to improve the accuracy of medical assessment by reducing its subjective nature and the time required by physicians for pathology diagnosis. This study is concerned with developing an automatic system platform where different PF ultrasound images can be visually improved, analysed and classified as normal or abnormal, using different medical image processing techniques, so as to provide more information to the doctors and the clinical treatment department for early diagnosis and the detection of the PF associated medical problems.

How Will The Result Data Be Protected?

The feedback from you and other medical experts will be anonymised. All information will be treated confidentially and data will be stored in a locked filing cabinet.

How Will The Result Data Be Used?

The result of the clinical evaluation will be used in assessing the automated classification system and it will also be introduced in the thesis.

Contact Information

If you have any queries or concerns regarding this study, please don't hesitate to contact us. **Thank you.**

Abdelhafid Boussouar
PhD Student, Informatics Research Centre,
School of Computing, Science and Engineering,
Newton Building,
M5 4WT, University of Salford,
Salford, UK.
+44 (0) 7746746938
a.boussouar1@edu.salford.ac.uk

Prof Farid Meziane
Head of Informatics Research Centre,
School of Computing, Science and Engineering,
Newton Building,
M5 4WT, University of Salford,
Salford, UK.
+44 (0) 161 295 3699
F.Meziane@salford.ac.uk

Dr Gillian Crofts
Senior Lecturer
School of Health Science,
Allerton Building,
M6 6PU, University of Salford,
Salford, UK.
+44 (0) 161 295 7021
G.Crofts@salford.ac.uk

Figure A.4: Participant Information Sheet (page 3)

Appendix B

Main Matlab GUI system and code for the despeckling evaluation study

The Speckle reduction GUI system implements several Despeckling techniques, as discussed in Chapter 2 and 3. This GUI framework supports a wide range of ultrasound image pre-processing functionalities such as: opening image files, ultrasound images pre-processing such as resizing, converting to grey-scale level, de-noising, feature extraction and despeckling evaluation using 11 selected metrics and display all the results in one window for each operation. It can also link with any MATLAB M-file using 'callback' functions.

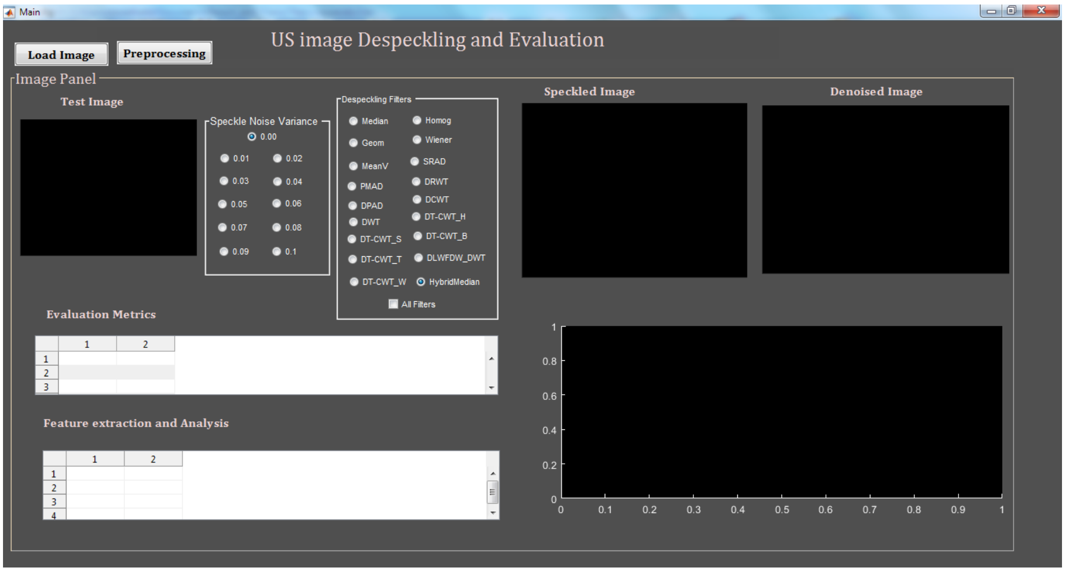


Figure B.1: Main Matlab GUI figure and for Despeckling evaluation study.

B.1 Matlab Code

Due to the limited space the following MATLAB Code represent only the evaluation part.

```

function varargout = Despeckling_Main_GUI(varargin)
%warning('off','all')

% Main M-file for Despeckling_Main_GUI.fig
% Begin initialization code -
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Main_OpeningFcn, ...
                   'gui_OutputFcn',  @Main_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
```

```
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code –


% —— Executes just before Main is made visible.
function Main_OpeningFcn(hObject, eventdata, handles, varargin)

handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT make Main wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% —— Outputs from this function are returned to the command line.
function varargout = Main_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved – to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global Image map
[filename pathname] = uigetfile('*','Select An Image');
[Image,map] = imread([pathname filename]);
axes(handles.axes1);
imshow(Image,map);
axis off
[m n c] = size(Image);
if ndims(Image) == 3;
    Image  = rgb2gray(Image);
end


% --- Executes when selected object is changed in GROUP BUTTON uipanel7.
function uipanel7_SelectionChangeFcn(hObject, eventdata, handles)

global Image noisy1 map
```

```

switch get(eventdata.NewValue,'Tag') % Get Tag of selected object.

    case 'speckle_0'
    noisy1 = Image;
axes(handles.axes24);
imshow(noisy1,map);
    case 'speckle_1'
noisy1 = imnoise(Image,'Speckle',0.01);
axes(handles.axes24);
imshow(noisy1,map);
    case 'speckle_2'
noisy1 = imnoise(Image,'Speckle',0.02);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_3'
noisy1 = imnoise(Image,'Speckle',0.03);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_4'
noisy1 = imnoise(Image,'Speckle',0.04);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_5'
noisy1 = imnoise(Image,'Speckle',0.05);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_6'
noisy1 = imnoise(Image,'Speckle',0.06);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_7'
```

```
noisy1 = imnoise(Image,'Speckle',0.07);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_8'
noisy1 = imnoise(Image,'Speckle',0.08);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_9'
noisy1 = imnoise(Image,'Speckle',0.09);
axes(handles.axes24);
imshow(noisy1);
case 'speckle_10'
noisy1 = imnoise(Image,'Speckle',0.1);
axes(handles.axes24);
imshow(noisy1);
end

% --- Executes when selected object is changed in uipanel_filtering.
function uipanel_filtering_SelectionChangeFcn(hObject, eventdata, handles)
global noisy1 Image  cnames cnames1
switch get(eventdata.NewValue,'Tag') % Get Tag of selected object.
%Mean Variance Filter
case 'filter_1'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im1= DsFlsmv (noisy1, [5 5], 5); %figure, imshow(outimage), title ('Despeckled image');
axes(handles.axes2);
imshow(filtered_im1);
```

```
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im1));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];
cnames = {'mean','st','skew','eng','ent' ,...%First Order Statistics (FOS) (5)
          'ang_sec_mom (eng)', 'cont', 'corr', 'sum_squ(var)','inv_diff_mom(hom)', 'sum_ave', 'sum_var', 'sum_ent' ,...
          'ent', 'diff_var', 'diff_ent', 'f12_inf_meas', 'f13_inf_meas' ,... %f12 f13 information measures of correlation ,...% Haralick Spatial Gray Level Dependence Matrices (SGLDM) (13)
          'hom','con','eng','ent','mean' ,...%Gray Level Difference Statistics (GLDS) (5)
          'fr','fa' ,... %Fourier Power Spectrum (FPS) (2)
          'h_mean','h_variance' ,...% Haar wavelet Features (2)
          'area','perim','MajorAxisLength','EquivDiameter','Extent','ConvexArea' ,... %shape features (6)
          };
rnames = {'Original image','MeanV','Fea_Diff'};
set(handles.uitable5 ,'Data',dat);
set(handles.uitable5 , 'ColumnName', cnames);
set(handles.uitable5 , 'RowName', rnames);

[Eval_metrics]=DsQmetrics(noisy1 ,filtered_im1);
dat1=[Eval_metrics];
```

```
cnames1 = {'MSE','SNR','RMSE','PSNR','ERR3','ERR4','UQI','SSIM','AD','CoC','EPI'};
rnames1 = {'MeanV'};
set(handles.uitable6,'Data',dat1);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', {'MSE','SNR','RMSE','PSNR','ERR3','ERR4','UQI','SSIM','AD','CoC','EPI'});
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
```

```
            set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
            % Place text atop the bar
            barTopper = sprintf(' %.3f', Eval_metrics(b));
            text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
            hold on;
end

%Wiener Filter
case 'filter_2'
%resize images
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im2= DsFwiener2(double(noisy1),[5 5]);
axes(handles.axes2);
imshow(filtered_im2);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im2));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','Wiener','Fea_Diff'};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im2);
dat2=[Eval_metrics];
```

```

rnames1 = {'Wiener'};
set(handles.uitable6,'Data',dat2);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel',
   cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4,
   3)
% Plot each number one at a time, calling bar() for each y
    value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(
            b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor',
            barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf('%.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end


%Median filter
case 'filter__2'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im2= DsFmedian(double(noisy1));
axes(handles.axes2);
imshow(filtered_im2);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im2));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','Median','Fea_Diff'};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im2);
dat2=[Eval_metrics];

rnames1 = {'Median'};
```

```
set(handles.uitable6,'Data',dat2);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end

case 'HybridMedian'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im2= DsFmedian_1(double(noisy1));
axes(handles.axes2);
imshow(filtered_im2);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im2));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','HybridMedian','Fea_Diff'};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im2);
dat2=[Eval_metrics];

rnames1 = {'HybridMedian'};
set(handles.uitable6,'Data',dat2);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

```

```
axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end
```

```

case 'Homog'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im2= DsFhomog(noisy1, [5 5], 3);
axes(handles.axes2);
imshow(filtered_im2);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\
    Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im2));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\
    Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','Homog','Fea_Diff'};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im2);
dat2=[Eval_metrics];

rnames1 = {'Homog'};
set(handles.uitable6,'Data',dat2);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
```

```
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end


case 'Geom'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
```

```
Image = bilinearInterpolation(Image, [512 512]);
filtered_im2= DsFgf4d (noisy1, [5 5], 5);
axes(handles.axes2);
imshow(filtered_im2);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im2));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','Geom','Fea_Diff'};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im2);
dat2=[Eval_metrics];

rnames1 = {'Geom'};
set(handles.uitable6,'Data',dat2);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%         'MarkerEdgeColor','m',...
%         'MarkerFaceColor','b')
```

```
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end

%PMAD filter
case 'filter_3'
        noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
    %   num_iter = 15;
%   delta_t = 1/7;
%    kappa = 30;
%    option = 2;
%    ad = anisodiff2D(s,num_iter,delta_t,kappa,option);
filtered_im3= DsFad(noisy1,5, 30, 0.25, 2);
axes(handles.axes2);
imshow(filtered_im3);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im3));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','PMAD','Fea_Diff'};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im3);
dat2=[Eval_metrics];

rnames1 = {'PMAD'};
set(handles.uitable6,'Data',dat2);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end
%SRAD filter
case 'filter_4'
    noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im4 = srad(noisy1,3);
```

```
axes(handles.axes2);
imshow(filtered_im4 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im4));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat4=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','SRAD','Fea_Diff'};
set(handles.uitable5,'Data',dat4);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im4);
dat_4=[Eval_metrics];

rnames1 = {'SRAD'};
set(handles.uitable6,'Data',dat_4);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
```

```
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end


   %DWT Filter
case 'filter_5'
%     dbstop if error
% im=imread('Image plantar fascia_Abnormal.bmp');
% im = imread('onion.png');
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
```

```
% figure;
% imshow(im);
% figure;
% imshow(out);
filtered_im5= denS2D(noisy1,20);
axes(handles.axes2);
imshow(filtered_im5 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\
    Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im5));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\
    Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat5=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DWT','Fea_Diff'};
set(handles.uitable5,'Data',dat5);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im5);
dat_5=[Eval_metrics];

rnames1 = {'DWT'};
set(handles.uitable6,'Data',dat_5);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
```

```
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel',
   cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4,
   3)
% Plot each number one at a time, calling bar() for each y
    value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(
           b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor',
           barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, '
           FontSize', barFontSize, 'Color','r');
        hold on;
end


%DRWT Filter
case 'filter_6'
```

```

%       dbstop if error
% im=imread('Image plantar fascia_Abnormal.bmp');
% im = imread('onion.png');
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
% figure;
% imshow(im);
% figure;
% imshow(out);
filtered_im6= denR2D(noisy1,20);
axes(handles.axes2);
imshow(filtered_im6,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im6));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat6=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DRWT','Fea_Diff'};
set(handles.uitable5,'Data',dat6);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im6);
dat_6=[Eval_metrics];

rnames1 = {'DRWT'};
set(handles.uitable6,'Data',dat_6);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf('%.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end


%DCWT Filter

case 'filter_7'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im7= denC2D(noisy1,20);
axes(handles.axes2);
imshow(filtered_im7,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\
    Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im7));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\
    Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat7=[original_im_feat;filtered_im_feat;fea_diff];
rnames = {'Original image','DCWT','Fea_Diff'};
set(handles.uitable5,'Data',dat7);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im7);
dat_7=[Eval_metrics];

rnames1 = {'DCWT'};
set(handles.uitable6,'Data',dat_7);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

```

```
axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end
```

```

% DT-CWT_H Filter
case 'filter_8'
%       dbstop if error
% im=imread('Image plantar fascia_Abnormal.bmp');
% im = imread('onion.png');
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
% figure;
% imshow(im);
% figure;
% imshow(out);
filtered_im8= denoising_dtdwt_hard(noisy1);
axes(handles.axes2);
imshow(filtered_im8 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im8));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat8=[original_im_feat;filtered_im_feat;fea_diff];
rnames = {'Original image','DT-CWT_H','Fea_Diff'};
set(handles.uitable5 ,'Data',dat8);
set(handles.uitable5 , 'ColumnName', cnames);
set(handles.uitable5 , 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im8);
dat_8=[Eval_metrics];

rnames1 = {'DT-CWT_H'};
set(handles.uitable6 ,'Data',dat_8);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel',
    cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4,
    3)
% Plot each number one at a time, calling bar() for each y
     value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(
            b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor',
            barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
```

```
            text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
            hold on;
end

%DT-CWT_S Filter
case 'filter_9'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
% figure;
% imshow(im);
% figure;
% imshow(out);
filtered_im9= denoising_dtdwt_soft(noisy1);
axes(handles.axes2);
imshow(filtered_im9 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im9));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat9=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DT-CWT_S','Fea_Diff'};
set(handles.uitable5 ,'Data',dat9);
set(handles.uitable5 , 'ColumnName', cnames);
set(handles.uitable5 , 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im9);
dat_9=[Eval_metrics];

```

```
rnames1 = {'DT-CWT_S'};
set(handles.uitable6,'Data',dat_9);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%       'MarkerEdgeColor','m',...
%       'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end

%DT-CWT_T Filter
case 'filter_10'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
alpha=7;
filtered_im10= denoising_dtdwt_trimmed(noisy1,alpha);
axes(handles.axes2);
imshow(filtered_im10 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im10));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat10=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DT-CWT_T','Fea_Diff'};
set(handles.uitable5,'Data',dat10);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im10);
dat_10=[Eval_metrics];

rnames1 = {'DT-CWT_T'};
set(handles.uitable6,'Data',dat_10);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%        'MarkerEdgeColor','m',...
%        'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end

%DT-CWT_B Filter
case 'filter_11'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im11= denoising_dtdwt_bivariate(noisy1);
axes(handles.axes2);
imshow(filtered_im11 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im11));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat11=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DT-CWT_B','Fea_Diff'};
set(handles.uitable5,'Data',dat11);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im11);
dat_11=[Eval_metrics];

rnames1 = {'DT-CWT_B'};
set(handles.uitable6,'Data',dat_11);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);
```

```

axes(handles.axes25);
grid off;
cla(handles.axes25);
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end

%DPAD Filter
case 'filter_12'
```

```
% im=imread('Image plantar fascia_Abnormal.bmp');
% im = imread('onion.png');
% noisy1 = bilinearInterpolation(noisy1, [512 512]);
% Image = bilinearInterpolation(Image, [512 512]);
% figure;
% imshow(im);
% figure;
% imshow(out);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\DPAD');
filtered_im12= dpad(noisy1,0.02,200,'cnoise',1,'big',1,'aja');
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\DPAD');
% filtered_im12=uint8(filtered_im12);
axes(handles.axes2);
imshow(filtered_im12 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im12));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat12=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DPAD','Fea_Diff'};
set(handles.uitable5 ,'Data',dat12);
set(handles.uitable5 , 'ColumnName', cnames);
set(handles.uitable5 , 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im12);
dat_12=[Eval_metrics];
```

```

rnames1 = {'DPAD'};
set(handles.uitable6,'Data',dat_12);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%        'MarkerEdgeColor','m',...
%        'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
```

```
            % Place text atop the bar
            barTopper = sprintf(' %.3f', Eval_metrics(b));
            text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
            hold on;
end

%DT-CWT_W Filter
case 'filter_13'
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im13= denoising_dtdwt_wiener(noisy1);
axes(handles.axes2);
imshow(filtered_im13 ,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im13));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat13=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DT-CWT_W','Fea_Diff'};
set(handles.uitable5,'Data',dat13);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im13);
dat_13=[Eval_metrics];

rnames1 = {'DT-CWT_W'};
set(handles.uitable6,'Data',dat_13);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%        'MarkerEdgeColor','m',...
%        'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.9);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end

case 'filter_14'

        %-------------Set the parameters of the denoising algorithm---------------------------
wav_base1='db4'; % the wavelet base used in the first LWFDW
wav_base2='db4';% the wavelet base used in the second LWFDW
level=5;          % the wavelet decomposition level
pad_mode=0;% 0:stands for the period extention mode
           % 1:stands for the symmetric extention mode

symbol=0; % 0: stands for denoising using Decimated wavelet transform
          % 1: stands for denoising using Undecimated wavelet transform

% If symbol=0, uncomment the following 4 code lines for reproducing the
% results of denoising image using maximal decimated wavelet transform
r1=[5,4,4,3,3];a1=2;  % the parameters of the directional window used in
%                       the first LWFDW under Decimated wavelet transform
r2=[3,2,2,1,1];a2=1.5;% the parameters of the directional window used in
%                       the second LWFDW
```

```
%———————Call the denoising function to denoise the noisy image————————

noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
tic;

filtered_im14=denoise_DLWFDW(noisy1,wav_base1,wav_base2,level,r1,a1...
                                        ,r2,a2,pad_mode,symbol);
toc;

%———————Compute the PSNR and show the result ——————————————————
axes(handles.axes2);
imshow(filtered_im14,[]);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat]=DsTTEXFEAT(double(filtered_im14));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff ]=original_im_feat-filtered_im_feat;
dat14=[original_im_feat;filtered_im_feat;fea_diff];

rnames = {'Original image','DLWFDW_DWT','Fea_Diff'};
set(handles.uitable5,'Data',dat14);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);

[Eval_metrics]=DsQmetrics(Image,filtered_im14);
dat_14=[Eval_metrics];

```

```
rnames1 = {'DLWFDW_DWT'};
set(handles.uitable6,'Data',dat_14);
set(handles.uitable6, 'ColumnName', cnames1);
set(handles.uitable6, 'RowName', rnames1);

axes(handles.axes25);
grid off;
cla(handles.axes25);
%bar(Eval_metrics);
% stem(Eval_metrics,'Marker','s',...
%        'MarkerEdgeColor','m',...
%        'MarkerFaceColor','b')
set(gca, 'XTick', 1:numel(Eval_metrics), 'XTickLabel', cnames1);
xlabel('Metrics');
ylabel('Metrics Values');
x = [1; 2; 3; 4; 5; 6; 7; 8; 9; 10; 11; 12; 13; 14; 15];
numberOfBars = length(Eval_metrics);
% Example of using colormap with random colors
barColorMap = rand(numberOfBars, 3);
%barColorMap(5:numberOfBars, 1:3) = rand(numberOfBars-4, 3)
% Plot each number one at a time, calling bar() for each y value.
barFontSize = 9;
for b = 1 : numberOfBars
        % Plot one single bar as a separate bar series.
        handleToThisBarSeries(b) = bar(x(b), Eval_metrics(b), 'BarWidth', 0.01);
        % Apply the color to this bar series.
        set(handleToThisBarSeries(b),'FaceColor', barColorMap(b,:));
        % Place text atop the bar
        barTopper = sprintf(' %.3f', Eval_metrics(b));
        text(x(b)-0.5, Eval_metrics(b)+2.8, barTopper, 'FontSize', barFontSize, 'Color','r');
        hold on;
end
figure(3);imshow(filtered_im14,[]);title('The denoised image using DLWFDW');
end

% --- Executes on button press in all_filters.
function all_filters_Callback(hObject, eventdata, handles)

global noisy1 Image  cnames cnames1
for z = 1:286
     %read input images in dataset folder
     outputBaseFileName = sprintf('image__%03d.png', z);
    %fullfile returns a string containing the full path to the file
    outputFullFileName = fullfile('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\All_data', outputBaseFileName);
    Image = imread(outputFullFileName);
    if ndims(Image) == 3;
    Image = rgb2gray(Image);
    end
    noisy1 = imnoise(Image,'Speckle',0.04);
noisy1 = bilinearInterpolation(noisy1, [512 512]);
Image = bilinearInterpolation(Image, [512 512]);
filtered_im1= median(double(noisy1));
filtered_im2= homog(noisy1, [5 5], 3);
filtered_im3= geom (noisy1, [5 5], 5);
filtered_im4= meanv (noisy1, [5 5], 5); %figure, imshow(outimage), title ('Despeckled image');
```

```
filtered_im5= wiener(double(noisy1),[5 5]);
filtered_im6= pmad(noisy1,5, 30, 0.25, 2);
filtered_im7= srad(noisy1,3);
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\DPAD');
filtered_im8= dpad(noisy1,0.02,200,'cnoise',1,'big',1,'aja');
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\DPAD');
filtered_im9= denS2D(noisy1,20);
filtered_im10= denC2D(noisy1,20);%%%%%%%%%%%%%%%%%10
filtered_im11= denoising_dtdwt_soft(noisy1);%%%%%%%%%%%%%%%%%%11
filtered_im12= denoising_dtdwt_hard(noisy1);%%%%%%%%%%%%%%%%%12
alpha=7;
filtered_im13= denoising_dtdwt_trimmed(noisy1,alpha);%%%%%%%%%%%%%%%%%13
filtered_im14= denoising_dtdwt_bivariate(noisy1); %%%%%%%%%%%%%%%%%14
% filtered_im15= denoising_dtdwt_wiener(noisy1);%%%%%%%%%%%%%%%%%%15
 wav_base1='db4';
wav_base2='db4';%
level=5; pad_mode=0;%
symbol=0; %
r1=[5,4,4,3,3];a1=2;
r2=[3,2,2,1,1];a2=1.5;
tic;
filtered_im15=denoise_DLWFDW(noisy1,wav_base1,wav_base2,level,r1,a1...
                    ,r2,a2,pad_mode,symbol);%%%%%%%%%%%%%%%%%16
```

```
toc;
filtered_im16= Hmedian_1(double(noisy1)); %%%%%%%%%%%%%%%%%17
addpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[original_im_feat]=DsTTEXFEAT(double(noisy1));
[filtered_im_feat1]=DsTTEXFEAT(double(filtered_im1));
[filtered_im_feat2]=DsTTEXFEAT(double(filtered_im2));
[filtered_im_feat3]=DsTTEXFEAT(double(filtered_im3));
[filtered_im_feat4]=DsTTEXFEAT(double(filtered_im4));
[filtered_im_feat5]=DsTTEXFEAT(double(filtered_im5));
[filtered_im_feat6]=DsTTEXFEAT(double(filtered_im6));
[filtered_im_feat7]=DsTTEXFEAT(double(filtered_im7));
[filtered_im_feat8]=DsTTEXFEAT(double(filtered_im8));
[filtered_im_feat9]=DsTTEXFEAT(double(filtered_im9));
[filtered_im_feat10]=DsTTEXFEAT(double(filtered_im10));
[filtered_im_feat11]=DsTTEXFEAT(double(filtered_im11));
[filtered_im_feat12]=DsTTEXFEAT(double(filtered_im12));
[filtered_im_feat13]=DsTTEXFEAT(double(filtered_im13));
[filtered_im_feat14]=DsTTEXFEAT(double(filtered_im14));
[filtered_im_feat15]=DsTTEXFEAT(double(filtered_im15));
[filtered_im_feat16]=DsTTEXFEAT(double(filtered_im16));
% [filtered_im_feat17]=DsTTEXFEAT(double(filtered_im17));
% [filtered_im_feat18]=DsTTEXFEAT(double(filtered_im18));
rmpath ('C:\Users\Abdelhafid\Documents\MATLAB\Despeckling_Evaluation\TextureFeatures')
[fea_diff1 ]=original_im_feat-filtered_im_feat1;
[fea_diff2 ]=original_im_feat-filtered_im_feat2;
[fea_diff3 ]=original_im_feat-filtered_im_feat3;
[fea_diff4 ]=original_im_feat-filtered_im_feat4;
[fea_diff5 ]=original_im_feat-filtered_im_feat5;
[fea_diff6 ]=original_im_feat-filtered_im_feat6;
[fea_diff7 ]=original_im_feat-filtered_im_feat7;
```

```
[fea_diff8 ]=original_im_feat-filtered_im_feat8;
[fea_diff9 ]=original_im_feat-filtered_im_feat9;
[fea_diff10 ]=original_im_feat-filtered_im_feat10;
[fea_diff11 ]=original_im_feat-filtered_im_feat11;
[fea_diff12 ]=original_im_feat-filtered_im_feat12;
[fea_diff13 ]=original_im_feat-filtered_im_feat13;
[fea_diff14 ]=original_im_feat-filtered_im_feat14;
[fea_diff15 ]=original_im_feat-filtered_im_feat15;
[fea_diff16 ]=original_im_feat-filtered_im_feat16;
% [fea_diff17 ]=original_im_feat-filtered_im_feat17;
% % [fea_diff18 ]=original_im_feat-filtered_im_feat18;

dat=[original_im_feat;filtered_im_feat1;fea_diff1;filtered_im_feat2;fea_diff2;filtered_im_feat3;fea_diff3;filtered_im_feat4;fea_diff4;filtered_im_feat5;fea_diff5;filtered_im_feat6;fea_diff6 ;...
  filtered_im_feat7;fea_diff7;filtered_im_feat8;fea_diff8;filtered_im_feat9;fea_diff9;filtered_im_feat10;fea_diff10;filtered_im_feat11;fea_diff11;filtered_im_feat12;fea_diff12;filtered_im_feat13 ;...
  fea_diff13;filtered_im_feat14;fea_diff14;filtered_im_feat15;fea_diff15;filtered_im_feat16;fea_diff16]; %;filtered_im_feat17;fea_diff17;filtered_im_feat18;fea_diff18
cnames = {'mean','st','skew','eng','ent' ,...%First Order Statistics (FOS) (5)
        'ang_sec_mom (eng)', 'cont', 'corr', 'sum_squ(var)','inv_diff_mom(hom)', 'sum_ave', 'sum_var', 'sum_ent' ,...
```

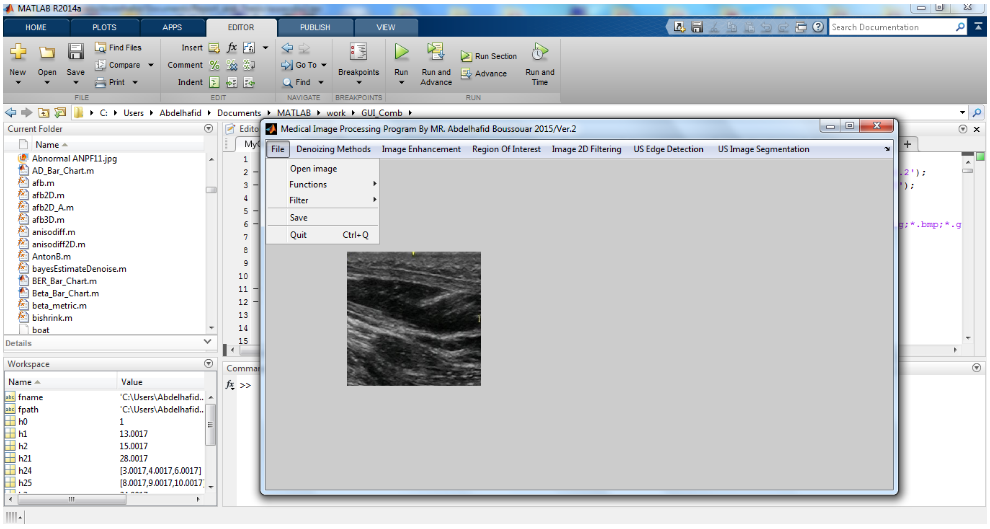
```
             'ent', 'diff_var', 'diff_ent', 'f12_inf_meas', '
                f13_inf_meas',... %f12 f13 information
                measures of correlation,...% Haralick Spatial
                Gray Level Dependence Matrices (SGLDM) (13)
             'hom','con','eng','ent','mean',...%Gray Level
                Difference Statistics (GLDS) (5)
             'fr','fa',... %Fourier Power Spectrum (FPS) (2)
             'h_mean','h_variance',...% Haar wavelet Features
                (2)
             'area','perim','MajorAxisLength','EquivDiameter',
                'Extent','ConvexArea',... %shape features (6)
              };
rnames = {'Original image','Median','Fea_Diff','Homog','
    Fea_Diff','Geom','Fea_Diff','MeanV','Fea_Diff','Wiener'
    ,'Fea_Diff','PMAD','Fea_Diff','SRAD','Fea_Diff','DPAD',
    'Fea_Diff','DWT','Fea_Diff','DRWT','Fea_Diff','DCWT','
    Fea_Diff','DT-CWT_S','Fea_Diff',...
     'DT-CWT_H','Fea_Diff','DT-CWT_T','Fea_Diff','DT-CWT_B'
       ,'Fea_Diff','DT-CWT_W','Fea_Diff','DLWFDW_DWT','
        Fea_Diff','HybridMedian','Fea_Diff',};
set(handles.uitable5,'Data',dat);
set(handles.uitable5, 'ColumnName', cnames);
set(handles.uitable5, 'RowName', rnames);
[Eval_metrics]=DsQmetrics(Image,noisy1);
[Eval_metrics1]=DsQmetrics(Image,filtered_im1);
[Eval_metrics2]=DsQmetrics(Image,filtered_im2);
[Eval_metrics3]=DsQmetrics(Image,filtered_im3);
[Eval_metrics4]=DsQmetrics(Image,filtered_im4);
[Eval_metrics5]=DsQmetrics(Image,filtered_im5);
[Eval_metrics6]=DsQmetrics(Image,filtered_im6);
[Eval_metrics7]=DsQmetrics(Image,filtered_im7);
[Eval_metrics8]=DsQmetrics(Image,filtered_im8);
[Eval_metrics9]=DsQmetrics(Image,filtered_im9);
```

```
[Eval_metrics10]=DsQmetrics(Image,filtered_im10);
[Eval_metrics11]=DsQmetrics(Image,filtered_im11);
[Eval_metrics12]=DsQmetrics(Image,filtered_im12);
[Eval_metrics13]=DsQmetrics(Image,filtered_im13);
[Eval_metrics14]=DsQmetrics(Image,filtered_im14);
[Eval_metrics15]=DsQmetrics(Image,filtered_im15);
[Eval_metrics16]=DsQmetrics(Image,filtered_im16);
% [Eval_metrics17]=DsQmetrics(Image,filtered_im17);
% % [Eval_metrics18]=DsQmetrics(Image,filtered_im18);


%Save despeckling results and feature extraction results
    to an excel file
%'evaluation_m1.xlsx'

fname='evaluation_m1.xlsx';
sname='Sheet1';
sname1='Sheet2';
sname2='Sheet5';
sname3='Sheet4';
sname4='Sheet5';
startingColumn='A';  %change if you want a different
    column
newData=[original_im_feat filtered_im_feat1
    filtered_im_feat2 filtered_im_feat3 filtered_im_feat4
    filtered_im_feat5 filtered_im_feat6 filtered_im_feat7
    filtered_im_feat8 filtered_im_feat9 filtered_im_feat10
    filtered_im_feat11 filtered_im_feat12
    filtered_im_feat13 filtered_im_feat14
    filtered_im_feat15 filtered_im_feat16
    filtered_im_feat17 filtered_im_feat18];
```

```
newData1=[fea_diff1 fea_diff2 fea_diff3 fea_diff4 fea_diff5 fea_diff6 fea_diff7 fea_diff8 fea_diff9 fea_diff10 fea_diff11 fea_diff12 fea_diff13 fea_diff15 fea_diff16 fea_diff17 fea_diff18];     %this our data
newData2= [Eval_metrics Eval_metrics1 Eval_metrics2 Eval_metrics3 Eval_metrics4 Eval_metrics5 Eval_metrics6  Eval_metrics7 Eval_metrics8 Eval_metrics9 Eval_metrics10 Eval_metrics11 Eval_metrics12 Eval_metrics13 Eval_metrics14 Eval_metrics15 Eval_metrics16]; %Eval_metrics17 Eval_metrics18
newData3=[weights weights1 weights2 weights3 weights4 weights5 weights6 weights7 weights8 weights9 weights10 weights11 weights12 weights13 weights14 weights15 weights16 weights17 weights18 ];
newData4=[ranked ranked1 ranked2 ranked3 ranked4 ranked5 ranked6 ranked7 ranked8 ranked9 ranked10 ranked11 ranked12 ranked13 ranked14 ranked15 ranked16 ranked17 ranked18 ];
[~,~,Data]=xlsread(fname,sname);  %read in the old data, text and all
nextRow=size(Data,1)+1;           %get the row number of the end
range=sprintf('%s%d',startingColumn,nextRow);  %this tells excel where to stick it
[~,~,Data1]=xlsread(fname,sname1);  %read in the old data, text and all
nextRow1=size(Data1,1)+1;           %get the row number of the end
range1=sprintf('%s%d',startingColumn,nextRow1);  %this tells excel where to stick it
[~,~,Data2]=xlsread(fname,sname2);  %read in the old data, text and all
```

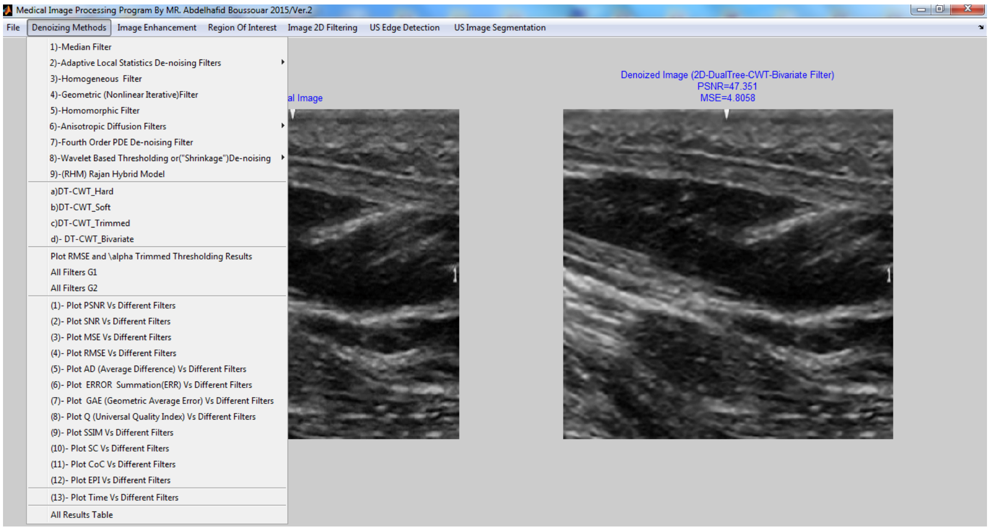
```
nextRow2=size(Data2,1)+1;            %get the row number of the end
range2=sprintf('%s%d',startingColumn,nextRow2);   %this tells excel where to stick it
[~,~,Data3]=xlsread(fname,sname3);   %read in the old data, text and all
nextRow3=size(Data3,1)+1;            %get the row number of the end
range3=sprintf('%s%d',startingColumn,nextRow3);   %this tells excel where to stick it
[~,~,Data4]=xlsread(fname,sname4);   %read in the old data, text and all
nextRow4=size(Data4,1)+1;            %get the row number of the end
range4=sprintf('%s%d',startingColumn,nextRow4);   %this tells excel where to stick it
xlswrite(fname,newData,sname,range);   % write the new data after the old data
xlswrite(fname,newData1,sname1,range1);
xlswrite(fname,newData2,sname2,range2);
xlswrite(fname,newData3,sname3,range3);
xlswrite(fname,newData4,sname4,range4);
end
```

B.2 PF US images Despeckle filtering GUI interface (using drop-down lists) created for visual inspection



Name	Value
fname	'C:\Users\Abdelhafid..
fpath	'C:\Users\Abdelhafid..
h0	1
h1	13.0017
h2	15.0017
h21	28.0017
h24	[3.0017,4.0017,6.0017]
h25	[8.0017,9.0017,10.0017

Figure B.2: Main Matlab based prototype system for visual inspection.


Medical Image Processing Program By MR. Abdelhafid Boussouar 2015/Ver.2
File
Denoizing Methods
Image Enhancement
Region Of Interest
Image 2D Filtering
US Edge Detection
US Image Segmentation
1)-Median Filter
2)-Adaptive Local Statistics De-noising Filters
3)-Homogeneous Filter
4)-Geometric (Nonlinear Iterative)Filter
5)-Homomorphic Filter
6)-Anisotropic Diffusion Filters
7)-Fourth Order PDE De-noising Filter
8)-Wavelet Based Thresholding or("Shrinkage")De-noising
9)-(RHM) Rajan Hybrid Model
a)DT-CWT_Hard
b)DT-CWT_Soft
c)DT-CWT_Trimmed
d)- DT-CWT_Bivariate
Plot RMSE and \alpha Trimmed Thresholding Results
All Filters G1
All Filters G2
(1)- Plot PSNR Vs Different Filters
(2)- Plot SNR Vs Different Filters
(3)- Plot MSE Vs Different Filters
(4)- Plot RMSE Vs Different Filters
(5)- Plot AD (Average Difference) Vs Different Filters
(6)- Plot ERROR Summation(ERR) Vs Different Filters
(7)- Plot GAE (Geometric Average Error) Vs Different Filters
(8)- Plot Q (Universal Quality Index) Vs Different Filters
(9)- Plot SSIM Vs Different Filters
(10)- Plot SC Vs Different Filters
(11)- Plot CoC Vs Different Filters
(12)- Plot EPI Vs Different Filters
(13)- Plot Time Vs Different Filters
All Results Table
al Image

Denoized Image (2D-DualTree-CWT-Bivariate Filter)
PSNR=47.351
MSE=4.8058

Figure B.3: Matlab based prototype system for visual inspection showing PF US images Despeckle filtering methods, origianal image and denoised image

B.3 PF US images Enhancement methods

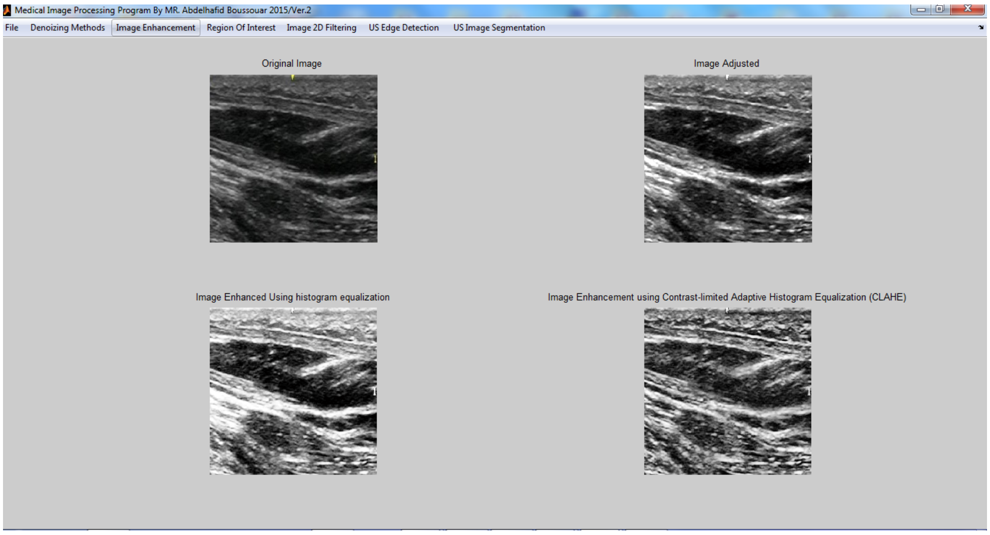

Medical Image Processing Program By MR. Abdelhafid Boussouar 2015/Ver.2
File
Denoizing Methods
Image Enhancement
Region Of Interest
Image 2D Filtering
US Edge Detection
US Image Segmentation
Original Image
Image Adjusted
Image Enhanced Using histogram equalization
Image Enhancement using Contrast-limited Adaptive Histogram Equalization (CLAHE)

Figure B.4: PF US images Enhancement methods

Appendix C

Main Matlab GUI interface for Segmentation, Classification and Novelty Detection approaches

The PF US images Segmentation, Classification and Novelty Detection GUI system implements the proposed techniques discussed in Chapter 4, 5 and 6. This GUI framework supports a wide range of ultrasound image pre-processing functionalities such as: display image files, pre-processing (resizing, converting to grey-scale level, de-noising, enhancement), segmentation, PF thickness estimation feature extraction, feature selection, PF classification (normal or abnormal), time execution, accuracy measures, and novelty detection results. It can also link with any MATLAB M-file using 'callback' functions.

C.1 Main GUI system for PF US images Segmentation, Classification and Novelty Detection tasks

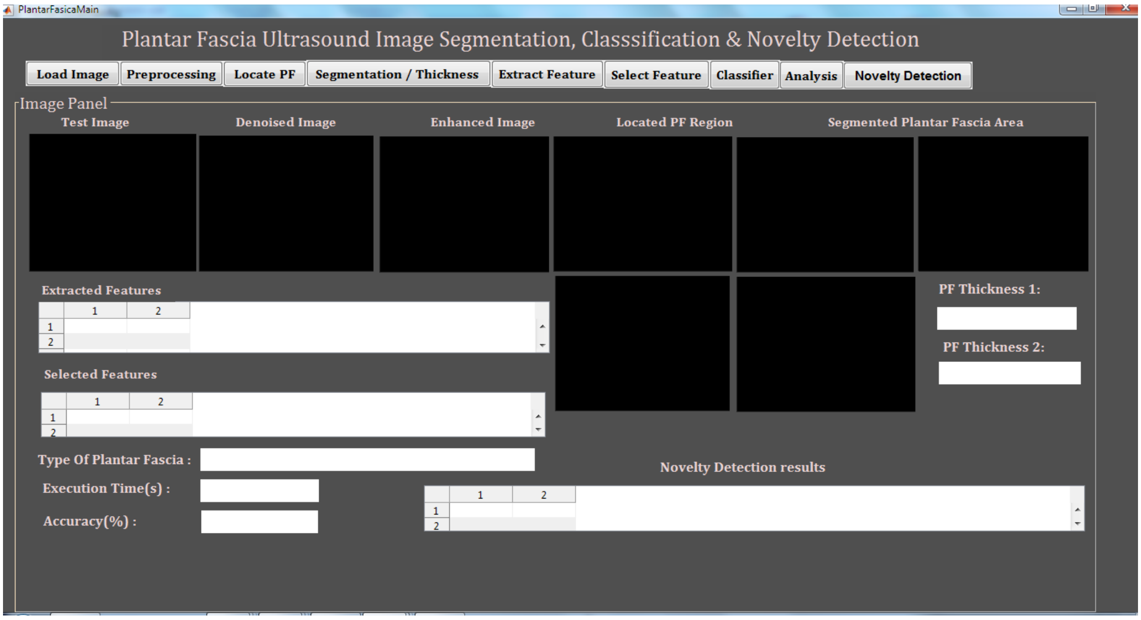
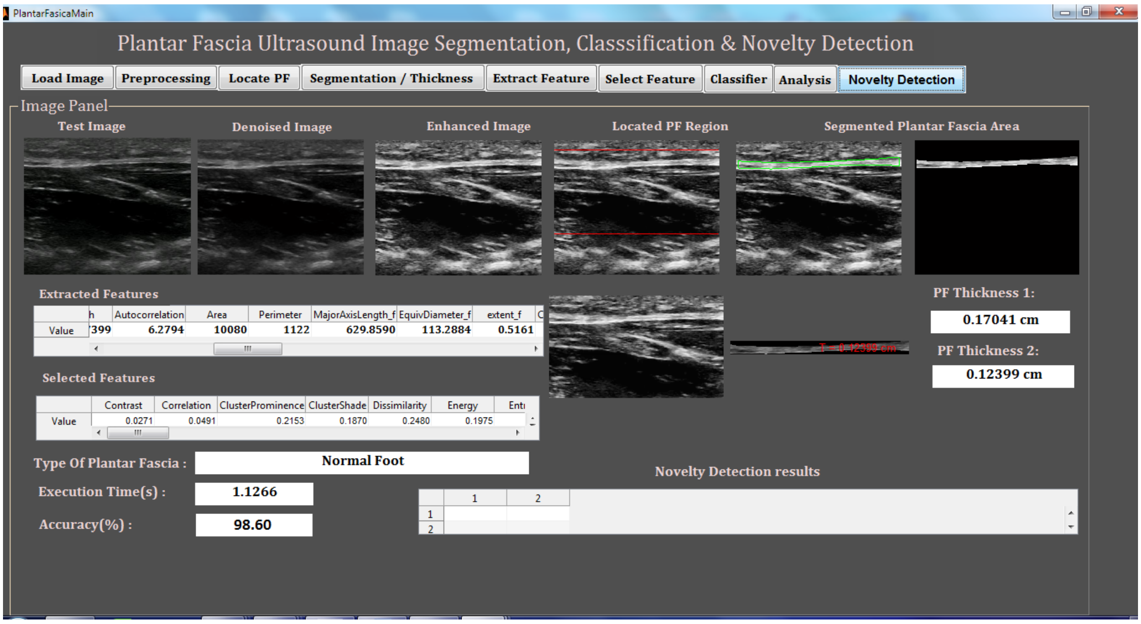


Figure C.1: Main Matlab GUI system for PF US images Segmentation, Classification and Novelty Detection tasks



Extracted Features

	h	Autocorrelation	Area	Perimeter	MajorAxisLength_f	EquivDiameter_f	extent_f	C
Value	'399	6.2794	10080	1122	629.8590	113.2884	0.5161	

Selected Features

	Contrast	Correlation	ClusterProminence	ClusterShade	Dissimilarity	Energy	Ent
Value	0.0271	0.0491	0.2153	0.1870	0.2480	0.1975	

PF Thickness 1: 0.17041 cm

PF Thickness 2: 0.12399 cm

Type Of Plantar Fascia : Normal Foot

Execution Time(s) : 1.1266

Accuracy(%) : 98.60

Figure C.2: Main Matlab GUI system showing PF US image segmentation and classification results using one classifier (knn)

C.2 Main Matlab Code example without evaluation part and callback functions due to the limited space

```
function varargout = PlantarFasciaMain(varargin)
%warning('off','all')
% PlantarFasciaMain M-file for PlantarFasciaMain.fig
% Begin initialization code
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn',
                      @PlantarFasciaMain_OpeningFcn, ...
                   'gui_OutputFcn',
                      @PlantarFasciaMain_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State,
       varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code

function PlantarFasciaMain_OpeningFcn(hObject, eventdata,
   handles, varargin)
handles.output = hObject;
guidata(hObject, handles);
```

```
function varargout = PlantarFasciaMain_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
global PlantarFascia_Image_1 PlantarFascia_Image filename
[filename, pathname] = uigetfile('*','Select An Image');

PlantarFascia_Image_1 = imread([pathname filename]);
PlantarFascia_Image=PlantarFascia_Image_1;
axes(handles.axes1);
imshow(PlantarFascia_Image);
axis off
[m n c] = size(PlantarFascia_Image);
if c == 3
    PlantarFascia_Image  = rgb2gray(PlantarFascia_Image);
end

function pushbutton2_Callback(hObject, eventdata, handles)
global PlantarFascia_Image
[ PlantarFascia_Image1 ] = Preprocess( PlantarFascia_Image );
axes(handles.axes2);
imshow(PlantarFascia_Image1);
[PlantarFascia_Image]=  adapthisteq (PlantarFascia_Image1);
axes(handles.axes5);
imshow(PlantarFascia_Image);
axis off
function pushbutton3_Callback(hObject, eventdata, handles)
global PlantarFascia_Image_1 segmentedImage PlantarFascia_Image

```

```

%****************** Segmentation *********************
addpath C:\Users\Abdelhafid\Documents\MATLAB\work\GUI_Comb
    \PlantarFasia_Segm\
[BW,segmentedImage,thicknesses,thick,blackMaskedImage]=
    PlantarFascia_Segmentation_Main (PlantarFascia_Image_1)
    ;
structBoundaries = bwboundaries(BW);
xy=structBoundaries{1};
x = xy(:, 2);
y = xy(:, 1);
axes(handles.axes3);
imshow(PlantarFascia_Image, []);
hold on;
plot(x, y, 'g', 'LineWidth', 1);drawnow;
axis off;
axes(handles.axes21);
imshow(segmentedImage);
hold on;
[lbl,N] = bwlabel(segmentedImage);
for idx = 1 : 1
         PF = (lbl == idx);
      [y,x]= find(PF);
      cen = mean([x y]);
      text(cen(1), cen(2), ['T = ' num2str(thicknesses(idx))
          ' cm'], 'color', 'red');
end
axis off
set(handles.text28,'String',[num2str(thick) ' cm']);
set(handles.text29,'String',[num2str(thicknesses) ' cm']);
blackMaskedImage = PlantarFascia_Image;
blackMaskedImage(~BW) = 0;
axes(handles.axes20);
```

```
imshow(blackMaskedImage)
axis off
rmpath C:\Users\Abdelhafid\Documents\MATLAB\work\GUI_Comb\
    PlantarFasia_Segm\

function pushbutton4_Callback(hObject, eventdata, handles)
global  PlantarFascia_Image TestImgFea segmentedImage
imagNew1 =segmentedImage;
%^^^^^^^^^^^^^^^^^^^^^Feature Extraction
    ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
     GLCM_mat = graycomatrix(imagNew1,'Offset',[2 0;0 2]);
     GLCMstruct = Computefea(GLCM_mat,0);
     v1=GLCMstruct.contr(1);
     v2=GLCMstruct.corrm(1);
     v3=GLCMstruct.cprom(1);
     v4=GLCMstruct.cshad(1);
     v5=GLCMstruct.dissi(1);
     v6=GLCMstruct.energ(1);
     v7=GLCMstruct.entro(1);
     v8=GLCMstruct.homom1(1);
     v9=GLCMstruct.homop(1);
     v10=GLCMstruct.maxpr(1);
     v11=GLCMstruct.sosvh(1);
     v12=GLCMstruct.autoc(1);
     %
         ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

      % Region based Features
      stats =regionprops(logical(imagNew1),'MajorAxisLength
          ','EquivDiameter','Extent','ConvexArea','
          Orientation');
        Area=length(find(imagNew1));
        Perimeter=sum(sum(bwperim(imagNew1,8)));
```

```
            MajorAxisLength_f = stats.MajorAxisLength;
            EquivDiameter_f=stats.EquivDiameter;
            extent_f =stats.Extent;
            ConvexArea_f =stats.ConvexArea;
            Orientation = stats.Orientation;
%       % Neighbourhood Gray Tone Difference Matrix (NGTDM)
             [coars1 ,contr ,busyn,compl,stren]=DsTNWNGTDMN(double(imagNew1),4);
%       %First Order Statistics (FOS)
         [mean,var,med,mode,skew,kurt,eng,ent]=DsTNWFOS(double(imagNew1));
%
% Statistical Feature Matrix (SFM)
[coars, cont, period, rough]=DsTNWSFM(double(imagNew1),4,4);
% feat=[feat, coars, cont, period, rough];

%Laws Texture Energy Measures (TEM)
[LL,EE,SS,LE,ES,LS]=DsTNWLAWS(double(imagNew1),7);
% feat=[feat, LL,EE,SS,LE,ES,LS];
TestImgFea = [v1,v2,v3,v4,v5,v6,v7,v8,v9,v10,v11,v12,Area,Perimeter,MajorAxisLength_f,EquivDiameter_f,extent_f,...
                              ConvexArea_f,Orientation, coars,contr,busyn,compl,stren,mean,var,med,mode,skew,kurt,eng,ent,coars1, cont, period, rough,LL,EE,SS,LE,ES,LS];
%
set(handles.uitable1,'Data',TestImgFea);
set(handles.uitable1, 'ColumnName', {'Contrast', 'Correlation','ClusterProminence','ClusterShade',....
        'Dissimilarity','Energy','Entropy','Homogeneity','Homop','Max.Prob' ,.....
        'Sosvh','Autocorrelation','Area','Perimeter','MajorAxisLength_f','EquivDiameter_f','extent_f','ConvexArea_f','Orientation' ,...
        'coars','contr','busyn','compl','stren','mean','var','med','mode','skew','kurt','eng','ent','coars1','cont', 'period','rough','LL','EE','SS','LE','ES','LS'});
set(handles.uitable1, 'RowName', {'Value'});

function pushbutton5_Callback(hObject, eventdata, handles)
global TestImgFea trainselectfea testselectfea PlantarFasciacate TrainImgFea
load TrainFeature
%*****************Feature Selection ************************
X = TrainImgFea;
y = PlantarFasciacate';
k=10;
cv = cvpartition (y, 'kfold' ,k);
opts = statset('display','iter');
 fun = @(XT,yT,Xt,yt)...
 (sum(~strcmp(yt,fitctree(XT,yT))));
[fs1,history] = infFS(X,0.9);
 fs = X(fs1(1:15));
trainselectfea = TrainImgFea;
testselectfea = TestImgFea;
msgbox('Feature selection completed');
set(handles.uitable3,'Data',fs);
 set(handles.uitable3, 'ColumnName', {'Contrast', 'Correlation','ClusterProminence','ClusterShade' ,....
        'Dissimilarity','Energy','Entropy','Homogeneity','Homop','Max.Prob' ,.....
        'Sosvh','Autocorrelation','Area','Perimeter','MajorAxisLength_f','EquivDiameter_f','extent_f','ConvexArea_f','Orientation' ,...
        'coars','contr','busyn','compl','stren','mean','var','med','mode','skew','kurt','eng','ent','coars1','cont', 'period','rough','LL','EE','SS','LE','ES','LS'});
set(handles.uitable3, 'RowName', {'Value'});
function pushbutton6_Callback(hObject, eventdata, handles)
global trainselectfea testselectfea PlantarFasciacate
load Truetype



[Imgcateind] = knnclassify(testselectfea,trainselectfea,PlantarFasciacate,5);
 switch(Imgcateind)
     case 1
         Imgcate = Truetype{Imgcateind,1};
     case 2
         Imgcate = Truetype{Imgcateind,1};
 end
 set(handles.text6,'String',Imgcate);

function pushbutton7_Callback(hObject, eventdata, handles)
global trainselectfea  PlantarFasciacate
Imgcate_whole = zeros(size(trainselectfea,1),1);
tic
for g = 1:size(trainselectfea,1)
    wholetestfea = trainselectfea(g,:);
    Imgcate_whole(g,1) = knnclassify( wholetestfea ,trainselectfea ,PlantarFasciacate);
end
endtime = toc;
set(handles.text8 ,'String',num2str(endtime));
[cmat grp] = confusionmat(PlantarFasciacate',Imgcate_whole);
figure(),
bar3(cmat);
set(gca, 'YTickLabel', {'Normal Foot', 'Diabetic Foot'});
set(gca, 'XTickLabel', {'Normal Foot', 'Diabetic Foot'});
title('Confusion Matrix')
%^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^

cpk = classperf(PlantarFasciacate',Imgcate_whole);
acc= cpk.CorrectRate;
acc = acc*100;
set(handles.text10 ,'String',num2str(acc));


function pushbutton8_Callback(hObject, eventdata, handles)
global PlantarFascia_Image

Iorg =PlantarFascia_Image;
I = imcrop(Iorg ,[10 10 700 300]);
I1 = Iorg;
s = sum(I1);
[notused, R1] = max(s);
[nttused, R2] = min(s);

axes(handles.axes6);
imshow(Iorg);
```

```
line(xlim, [30 30], 'Color', 'r'),
line(xlim, [300 300], 'Color', 'r'),
axes(handles.axes7);
imshow(I);
axis off

function pushbutton9_Callback(hObject, eventdata, handles)
NDtype = {'PARZEN'; 'GMM';'GPOC';'SOM';'SVDD'};
whichdata1 = 'PF_TrainFeature';
    paramsPlot.plotROC1 = true;
    paramsPlot.plotoutput1 = true;
        paramsPreset1 = [];
addpath C:\Users\Abdelhafid\Documents\MATLAB\NDtoolv0.12\;
 [machine1, outputMisc1, outputConf1, outputROC1,
    outputData1] = Main_ND(whichdata1, NDtype1,
    paramsPreset1, paramsPlot1);

if nargin < 1

    % ---- PF US  database ----
        whichdata = 'TrainFeature';
end

if nargin < 2
       NDtype = {'parzen'; 'gmm';'gpoc';'som';'SVDD'};
end

if nargin < 3
    paramsPreset = [];
end

if nargin < 4

```

```
    paramsPlot.plotROC1 = true;
    paramsPlot.plotoutput1 = true;
end


startupND;

%% # of ND methods
if iscell(NDtype)
    N = length(NDtype);
else
    N = 1;
    NDtype = {NDtype};
end
data = load(whichdata);
fprintf('\nLoading data set %s...\n', whichdata);
alldataOri1 = data.TrainImgFea; % numdata by numftrs
classlabels = data.PlantarFasciacate; % numdata by 1, class labels = 1, 2
classlabels=classlabels';


%% split normal data from abnormal data .
if length(unique(classlabels)) > 1 % more than one class labels; treat class 1 is set to be 'normal'
    isnor = classlabels == 1; % regard class 1 as normal.
    isab = ~isnor;
    normaldataOri1 = alldataOri1(isnor);
    abnormaldataOri1 = alldataOri1(~isnor);

    if isempty(normaldataOri1)
        error('Can not find class lable == 1; please re-define "normal" data.\n');
```

```matlab
    end

    if isempty(abnormaldataOri1)
        error('Abnormal data (class lable ~= 1) do not exist; all data are regarded as "normal" data.\n');
    end

    % Split and normalize dataset into training, validation, and test
    [traindataNorOri1, testdataNorOri1, validdataNorOri1, validdataAbOri1, testdataAbOri1] = splitData(alldataOri1, isab);
    % Train the ML classifier and get the output, for all the N methods.
    for i = 1 : N
        fprintf('\n');
        fprintf('--- one-class classification by %s approach ---\n', NDtype{i});
        [machine1, outputMisc1, outputConf1, outputROC1, outputData1,perf_eval] = runNoveltyDetection(traindataNorOri1,...
        validdataNorOri1, testdataNorOri1, validdataAbOri1, testdataAbOri1, lower(NDtype{i}), paramsPreset1, paramsPlot1);
    end
else
    fprintf('\nA dataset needed to test the classifiers performance!\n');
end
diary off % end of function
rmpath C:\Users\Abdelhafid\Documents\MATLAB\NDtoolv0.12\
dat1=perf_eval;
```

```
cnames1 = {'B-Accuracy','F-Score','MCC','Gmean','AUC','Time'};
rnames1 = {'Evaluation Metrics'};
set(handles.uitable4,'Data',dat1);
set(handles.uitable4, 'ColumnName', cnames1);
set(handles.uitable4, 'RowName', rnames1);


% write classifiers' evaluation results to spread sheet file.
fname='ND_perf_eval_results.xlsx';
 sname='Sheet1';
startingColumn='A';
 newData=perf_eval;
[~,~,Data]=xlsread(fname,sname);  %read in the old data, text and all
nextRow=size(Data,1)+1;             %get the row number of the end
range=sprintf('%s%d',startingColumn,nextRow);  %this tells excel where to stick it
 xlswrite(fname,newData,sname,range);  %
```

C.3 PF US images Segmentation Results for visual inspection

C.3.1 PF US images preprocessing

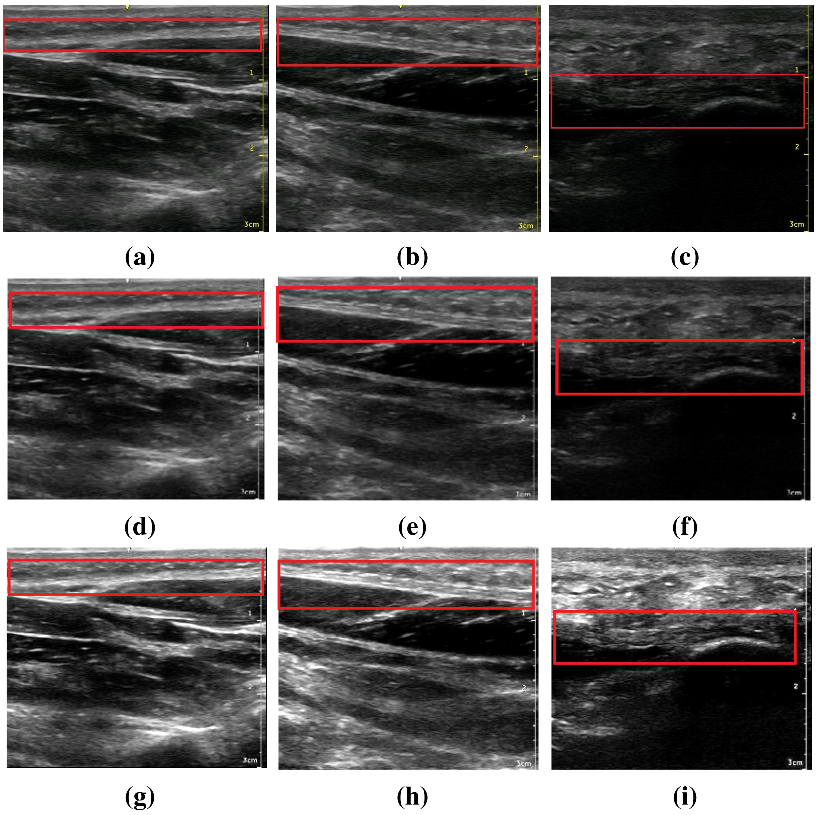
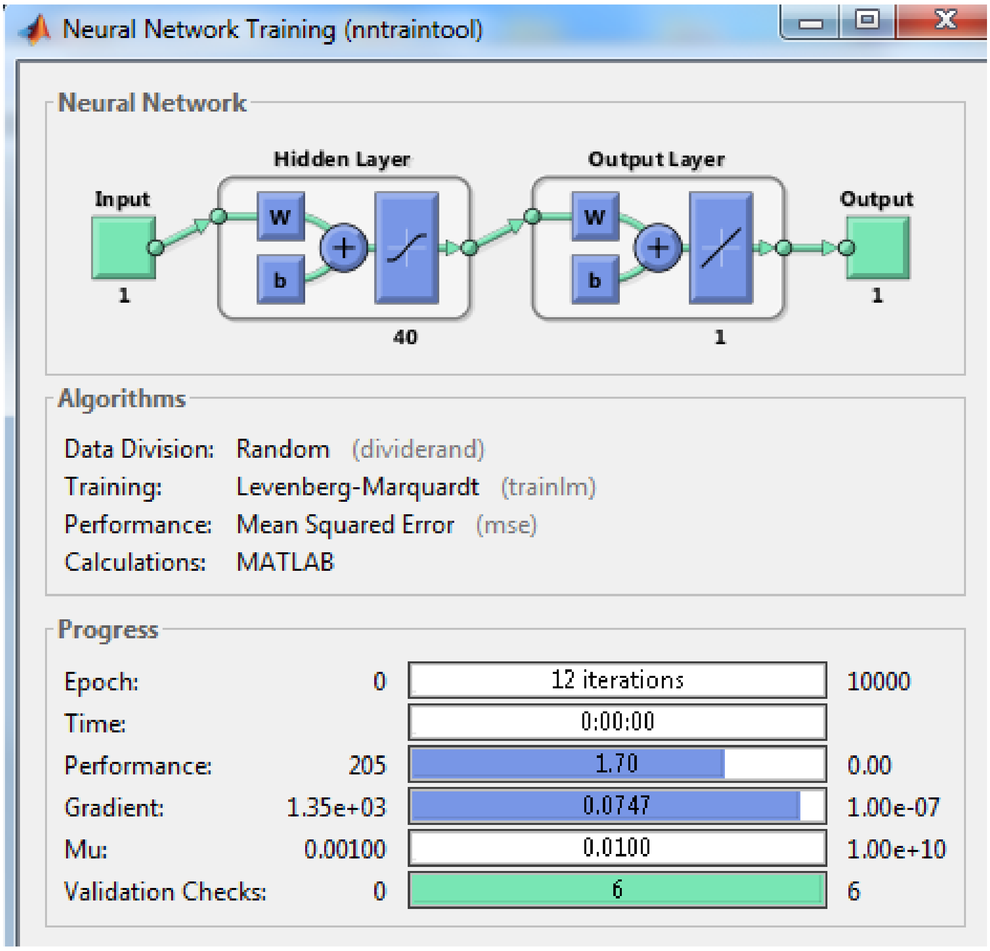


Figure C.3: Preprocessing results: (a)-(c) Original US images for different PF structures (Forefoot, Mid and Rear section). (d)-(f) De-speckling results using DT-CWT filter. (g)-(e) Enhancement results using CLAHE filter.

C.3.2 Training phase using RBF Neural Network



Algorithms

Data Division: Random (dividerand)
Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Calculations: MATLAB

Progress

Epoch:	0	12 iterations	10000
Time:		0:00:00	
Performance:	205	1.70	0.00
Gradient:	1.35e+03	0.0747	1.00e-07
Mu:	0.00100	0.0100	1.00e+10
Validation Checks:	0	6	6

Figure C.4: Training phase using RBF Neural Network

C.3.3 RBF Neural Network Validation Performance

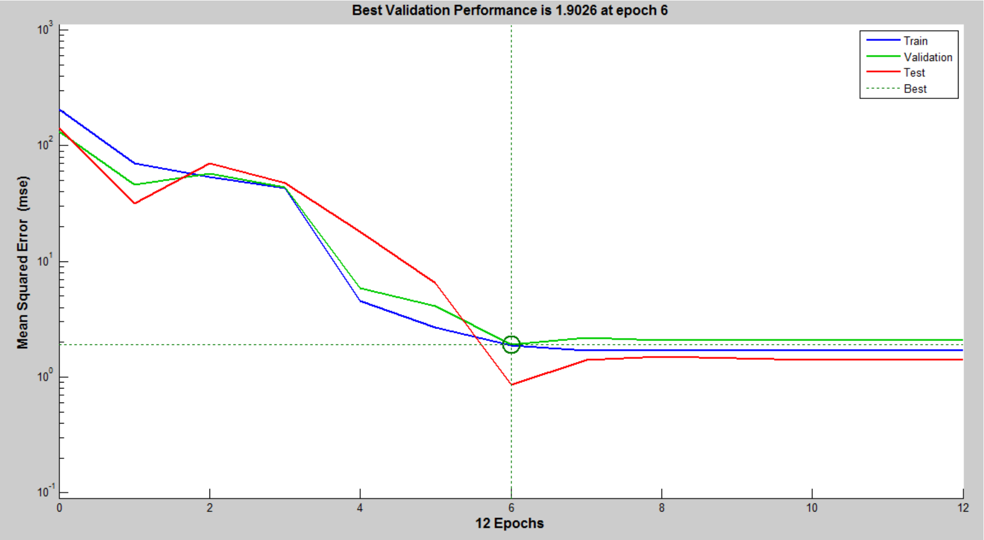


Figure C.5: RBF Neural Network Validation Performance using MSE metric

C.3.4 RBF Neural Network Regression graph

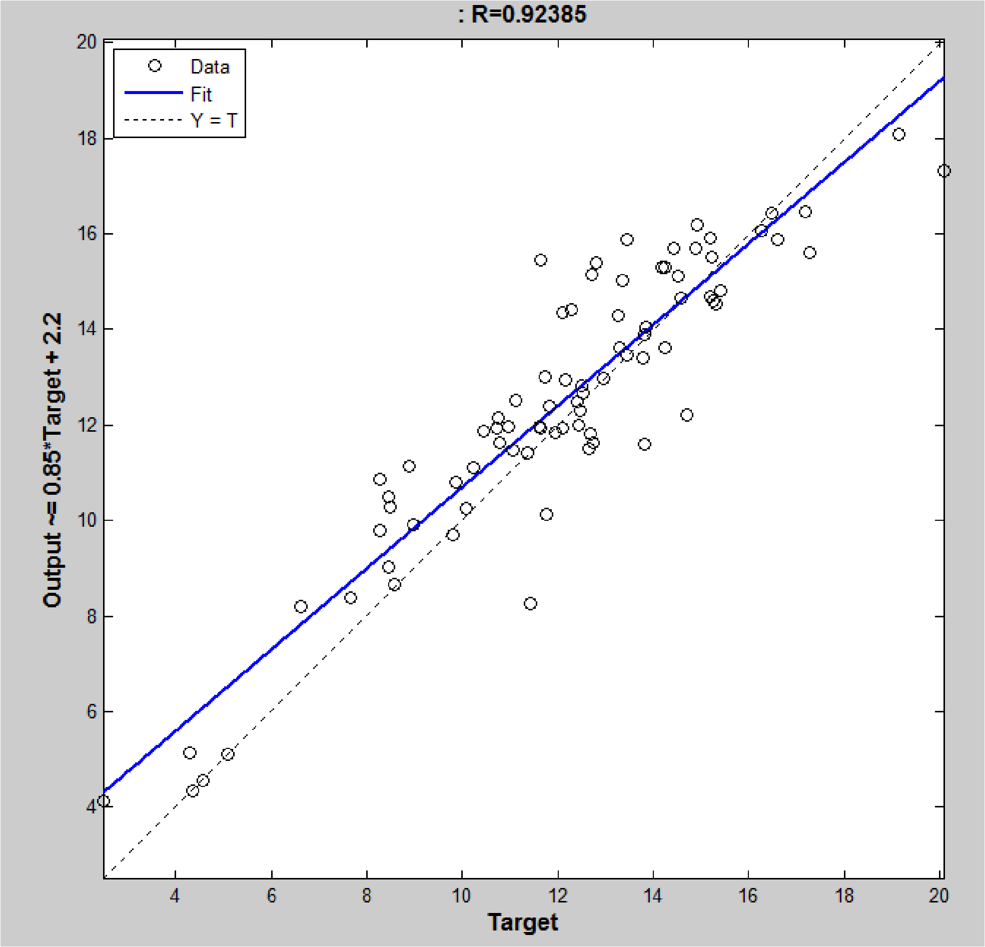
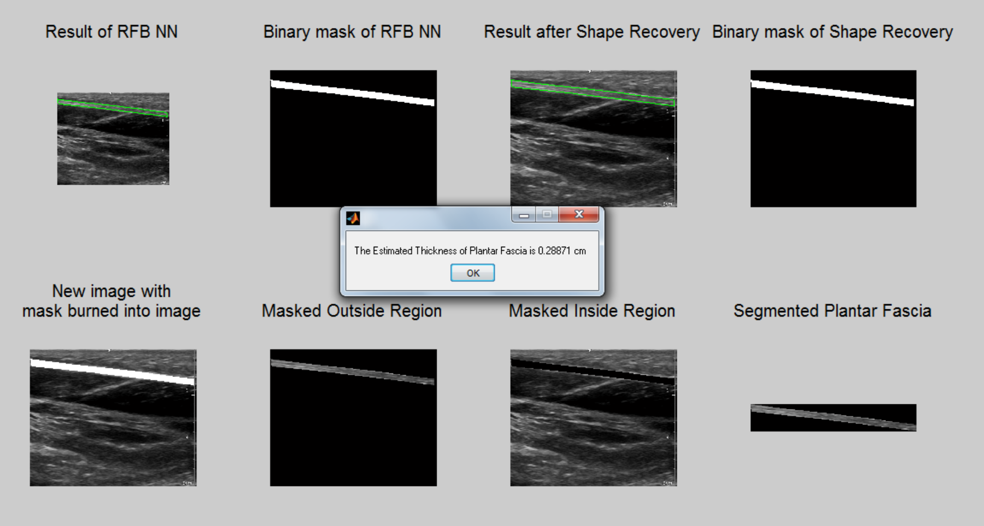


Figure C.6: RBF Neural Network Regression graphs

C.3.5 RBF Neural Network Segmentation results with estimated PF thickness



The Estimated Thickness of Plantar Fascia is 0.28871 cm

Figure C.7: RBF Neural Network Segmentation results with estimated PF thickness

C.4 Graph illustration of the output of different supervised machine learning methods used in Chapter 5

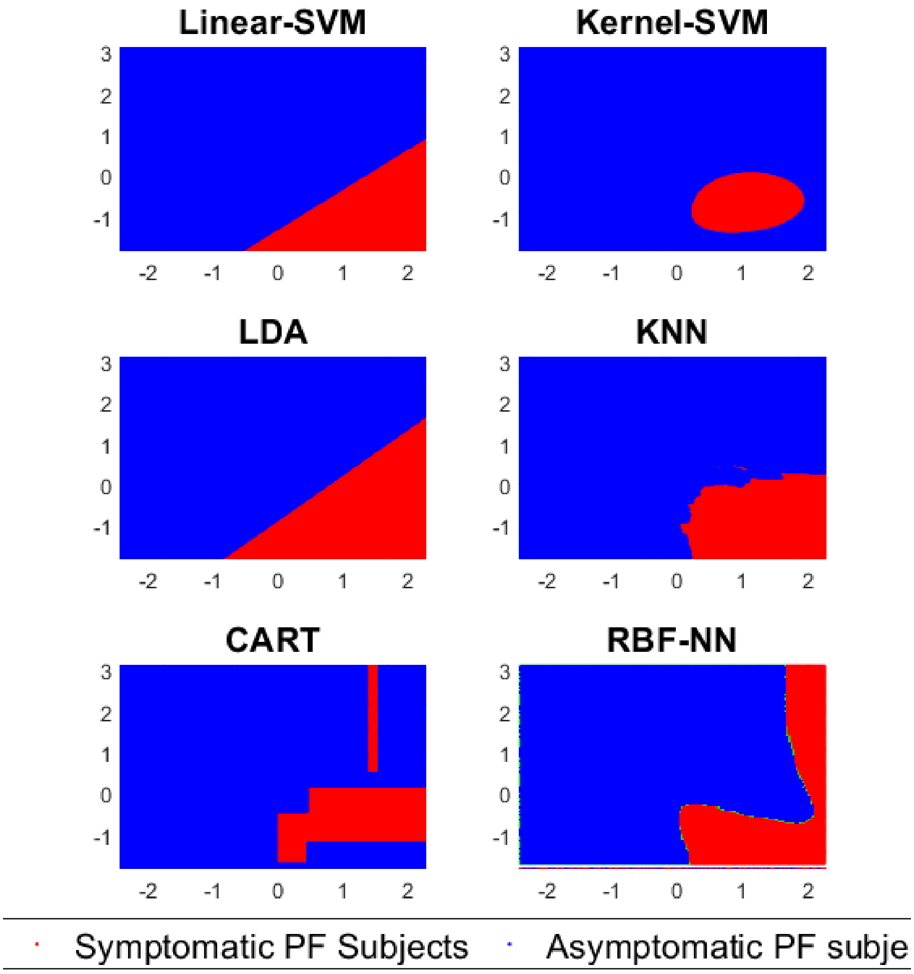


Figure C.8: Graphical illustration of all classification models as presented in Chapter 5

C.5 Graph plot showing the original 2-D features and the scaled (normalized) 2-D features

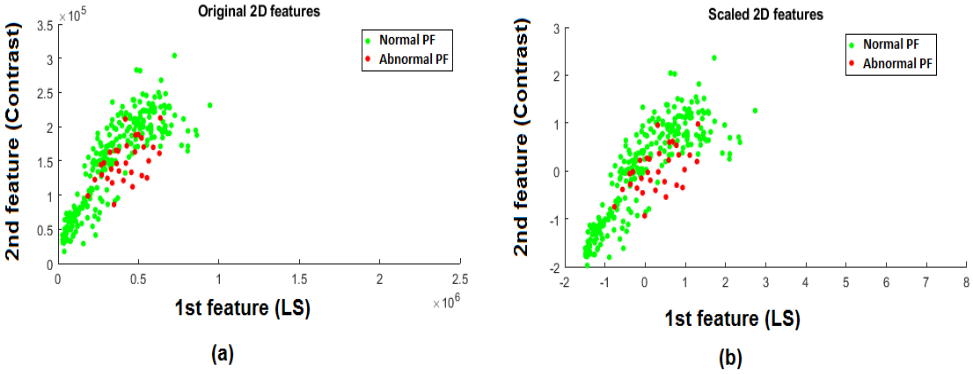


Figure C.9: Graph plot showing the original 2-D features and the scaled (normalized) 2-D features