Exploring the Possibilities in Automated Choroidal Diseases Classification

Dominic Chim
MSc in Advance Computer Science

Xianghua Xie, (Supervisor)

February 26, 2021

Abstract

The report aims to explore the possibility of creating a classifier for detecting the different stages of Age-Related Macular Degeneration with the potential impact to create an automated diagnosis in a real life environment. This has led to the research in areas related to Texture Analysis, Retinal Image Analysis and Machine Learning. This is inspired by analysing the goals and the problems presented as being similar in nature to a textual analysis problem.

We present an approach that is used to solve several stages of this experiment. We propose techniques used to handle the input data namely, Optical Coherence Tomography images via a labelling process which is supplied with support material on treating these types of data and what criterion was used to produce our data sample. An algorithm is developed to use an adaptation of a Gabor Filter, as one part of the experiment to produce texture features The Gabor Filter Banks is adapted mildly to suit the experiment otherwise, the bank of Gabor filters has remained standard thus uniformly covers the spatial-frequency domain. Texture features are then extracted by applying a non linear transform to the filtered images and computing a measure of energy at particular window sizes. Details on the implementation of this algorithm are noted in the implementation section of this report along with in depth support material.

We also developed two algorithms, one is developed in conjunction with several traditional classifiers such as k-Near Neighbour,Random Forest and Support Vector Machine, and a couple of the Deep Learning Algorithm, The Restricted Boltzmann Machine and Deep Belief Network. In order to produce a test suite for training and validating these classifiers with the acquired texture features as both the training and testing sample. In order to, measure the classifier's accuracy against patients wit different stages of Age-Related Macular Degeneration. The other algorithm is developed to compile the results from the classifiers and produce the overall classification result. An analysis of these result have provided both promising results and raised several important questions related to the progression of an automated classifier,which are discussed in the analysis section of this report.

Acknowledgements

I would like to thank the following: Dr Xianghua Xie, my supervisor, for his encouragement, understanding and expert advice; Jingjing Deng, RA, mentor and friend, for being very patient and understanding, providing motivation and assistance in many areas; Members of the Swansea University SwanseaVision Team, for providing insight and advice on matters related to both the project and the culture within research; and finally special thanks go to my friend and family for their support through my research. Special thanks to Cardiff University, for the allowing use of the Optical Coherence Tomography (OCT) machine and the OCT image provided, to which our entire project is based upon.

Contents

1 Introduction 5

1.1 Texture Analysis . . . 6

1.2 Retinal image Analysis . . . 7

1.2.1 Classification of AMD . . . 8

1.2.2 Scanning the Choroid . . . 10

1.3 Project Specification . . . 11

1.4 Aims and Objectives . . . 11

1.5 Motivation . . . 12

1.6 Materials . . . 12

2 Literature Review 13

2.1 Situation . . . 13

2.2 Texture Analysis . . . 14

2.2.1 Statistical Approaches . . . 14

2.2.2 Structural approaches . . . 14

2.2.3 Filter based approaches . . . 14

2.2.4 Spatial and Frequency domain filtering . . . 14

2.2.5 Joint Spatial/Spatial-frequency methods . . . 15

2.2.6 Model Based Approaches . . . 15

2.3 Classification . . . 16

2.3.1 Machine Learning . . . 16

2.3.2 K-Nearest Neighbour (k-NN) . . . 17

2.3.3 Random Decision Forest (RDF) . . . 17

2.3.4 Support Vector Machines . . . 20

2.4 Deep Learning . . . 22

2.4.1 Restricted Boltzmann Machines (RBM) . . . 22

2.4.2 Deep Belief Network (DBN) . . . 24

2.5 Choroidal Diseases and Disorders . . . 25

2.5.1 Age Related Macular Degeneration (AMD) . . . 26

2.5.2 Causes . . . 26

2.6 Optical Coherence Tomography (OCT) 27
2.6.1 History of OCT 28
2.6.2 Principles 29
2.6.3 Resolution 32
2.6.4 Types of OCT 34
2.6.5 OCT Improvements: Enhanced Depth Imaging (EDI) 38
2.6.6 Conclusion and Application 38

3 Implementation 39
3.1 Proposed Solution 39
3.2 Methodology 39
3.3 Labelling 40
3.3.1 Labelling Principles 40
3.3.2 Labelling 42
3.4 Image Processing 43
3.4.1 Masking 43
3.4.2 Filtering 44
3.4.3 Gabor Filters 44
3.5 Feature Extraction 45
3.6 Training and Classification 46

4 Results 48
4.1 Analysis 51
4.1.1 10 Fold 52
4.1.2 2 Fold 53
4.1.3 Leave-One-Patient-Out (LOVop) 54
4.2 Possible Refinements 55
4.2.1 Higher Quality Images 55
4.2.2 Adjustment of image window size 55
4.2.3 Utilising different Scanning Methods 56
4.2.4 New Features 57
4.2.5 Labelling Criterion 57
4.2.6 More Patients 58

5 Conclusion 59

Appendices 69

List of Figures

1.1 Layout of a human eye 7
1.2 Stages of Age-Related Macular Degeneration 9

2.1 Comparison between Choroidal Diseases 26
2.4 Schematic of OCT System Based on Michelson Interferometer [Ali and Parlapalli 2010] 31
2.5 Diagram showing a sample of the coherence length (Δt) 33
2.6 Diagram showing a typical SD-OCT setup [Podoleanu 2014] 35
2.7 Diagram showing a typical SS-OCT setup [Podoleanu 2014] 36
2.8 (a) Spectrum output of the interferometer for two OPD values, left: 3 cl, right: 6 cl; (b) SD-OCT sampling function, of width , spectrum is read in a time T either by downloading the charge from the linear camera array, or by tuning the frequency of the SS, the large arrow suggests that by doing so, the spectrum shown above in (a) is transferred to the output signal shown in (c) below; (c) signal delivered by the Processing unit in Figures 2.6 and 2.7. [Podoleanu 2014] 37

3.1 OCT Scan displaying the appearance retinal atrophy. 41
3.2 OCT Scan of Dry AMD 41
3.3 OCT Scan of PED & CNV 41
3.4 OCT scan with sample ROI highlighted 43

4.1 Grid of filter responses of a OCT image 49
4.2 Heat map of Filter Reponse 50
4.3 Comparision between new and old OCT images 56

List of Tables

3.1 A simplified example of expected Confusion Matrix 47

4.1 10 Fold Results 52
4.2 2 Fold Results 53
4.3 Leave-one-Patient-out Results 54

Chapter 1

Introduction

Texture analysis has been a popular field of interest, as it allows the analysis of various surface e.g. wood, metals even non flat objects such as fruit, without tampering with the surface in non-invasive manner Which as a result it is highly demanded by industry as it replaces the subjective and inefficient process of manual inspection. A particular example is mentioned in [Xie 2008], of texture analysis in the production of ceramic tiles, where due to external factors such as colour pigments, humidity, and temperature. An on-line automated monitoring system with adaptive feedback controls are more desirable, as the complexities of such systems start to increase.this report explores the possibilities of using texture analysis to detect choroidal diseases and will then go on to implement and train a classifier to detect and identify various choroidal diseases in an automated fashion. The final section will determine the quality of classification provided by the classifier and discuss future steps to be had in the project. Texture analysis would be applied to detect diseases in the choroid of the retina closely associated with the eye. The project will focus on the application of texture analysis and would be applied in conjunction with computer vision techniques such as using supervised learning models and other feature selection methods, in order to achieve the highest accuracy in classifying the correct form of choroid disease. Which as a result would allow users in medicine i.e. doctors and nurses to easily visualise the raw data, into a format that is more intuitive for them to understand, leading to more concise diagnoses regarding diseases in the retina.

1.1 Texture Analysis

The initial element of texture analysis often involves a texture and/or colour analysis followed by a form of pattern classification that leads to the system's decision making. The texture/colour analysis revolves around feature representation and extraction, as well as modelling the extracted data and visualising it. Pattern classification involves pattern representation, cluster analysis and discriminate classification [Xie 2008]. We will briefly discuss the texture feature extraction and analysis in 4 categories namely: statistical methods, structural methods, filter-based methods and model based methods.

However, bear in mind texture analysis should not be considered the same as general texture segmentation (the process of forming regions which can be used to extract particular regions) and classification (the processes of using processed data to intelligent formulate decisions based on a set criterion). As both defect-free and defective areas of inspected surfaces can be texturally unstable, i.e. they will be often further segmented into smaller regions while in defect detection the defective region should be treated as a whole no matter how further segmented the surface may be.[Xie 2008]

Classifying surfaces is also different from texture classification, as defective samples may not necessarily form a single class and the defect types may only be partially taken in account beforehand. Additionally, in some applications false positives (rejecting good samples) are more forgiveable than false negative (missing defective regions or samples). An example of this is the detection of tumours which the detection of cells that are benign (see glossary for definition) are much more appreciative than not finding potential cancerous cells.

A significant differentiating factor in the classification process is supervised classification versus novelty detection, in applications where both normal and defective samples can be easily obtained and pre-defined such as common medical data such as X-ray scans where defective areas can be highlighted, supervised classification based approaches are typically favoured. However, when defects are unpredictable and defective samples are unavailable such as samples shown in an aggressive mutating virus, novelty detection is more desirable. (For a more in-depth review of texture analysis please refer to A Review of Recent Advances in Surface Defect Detection using Texture analysis Techniques by Xianghua Xie [Xie 2008])

1.2 Retinal image Analysis

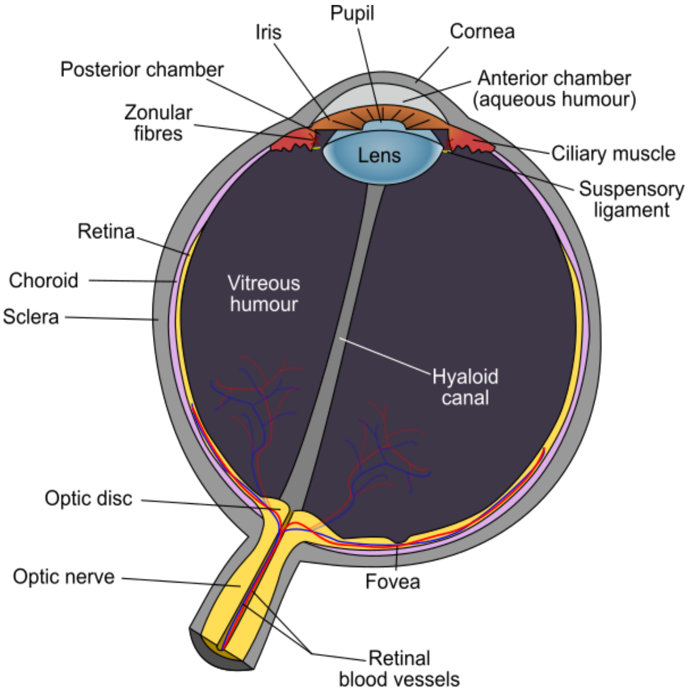


Figure 1.1: Layout of a human eye

Another major theme of the project is retinal image analysis, specifically the analysis of choroid diseases. The retina is the only locals where blood vessels can be directly visualised in a non-invasive form. Recently technology has allowed the development in digital imaging systems, whilst the resolution of these digital images pale in comparison to conventional photography. Modern digital imaging systems provide very high-resolution images that are sufficient for medical use [Patton et al. 2006]. In addition, digital images provide the advantage of easier storage non-deteriorating in quality media, and easily transferable in short distances within the clinic or across long distances thanks to electronic transfers through the use of the Internet. [Patton et al. 2006].

Finally, and more importantly these digital media can be subjected to image analysis for a particular criterion (which in this case in this case is the characteristics of choroid diseases), with the potential for an automated diagnosis. This is particular useful in research or screen setting where large databases of particular images (choroid scans in our case) may be automatically classified and manger more readily than labour intensive observer-driver techniques. As a result as previously mentioned automated diagnosis may also be used to

aid in the decision making for optometrists. The choroid is a pigmented highly vascular layer of the globe (the eyeball) that lies between the retina layer and the sclera layer (see figure 1.1 for a graphical visualisation). The main function of the choroid is to nourish the outer layer of the retina, with some thought that the choroid may also be used to regulate the heat that comes off the retina. An example of its use is that without the choroid supplying oxygen and nutrients through the blood within its vessels to the surrounding layers then the eye will suffer namely, the sclera would begin to dry out become inflamed and form ulcers, whilst the retina will not be able to process the light that it receives into messages for the brain.[Kokkinakis 2012] In summary the choroid is an essential part of the highly complex structure that is the human eye.
As a result, the choroid is subject to a variety of diseases and disorder. In practice both the term disease and disorder are in interchangeable as a form of respect for patients, however in terms of definition there is a slight difference in their usage:

- Disease thought of something you catch randomly from someone or something, such as bacterial or viral conjunctivitis.
- Disorder something that is wrong with the eyes such as near sightedness or colour blindness. Disorder infers perhaps that of something being merely a chance happening.

Due to the complex workings of the human eye, there are countless forms of diseases and disorders that affect choroid. We will discuss four particular choroid disease namely, age related macular degeneration (AMD) both the wet and dry variations, and Pigment Epithelial Detachments (PED) that occurs with Age-Related Macular Degeneration. The rest of the paper is organised such that in Aims and Objectives section we will discuss, how the aforementioned targets would be achieved, followed by discussing the purpose of this project and the impact it would have, which are mentioned in the
textbf*Motivations* section.

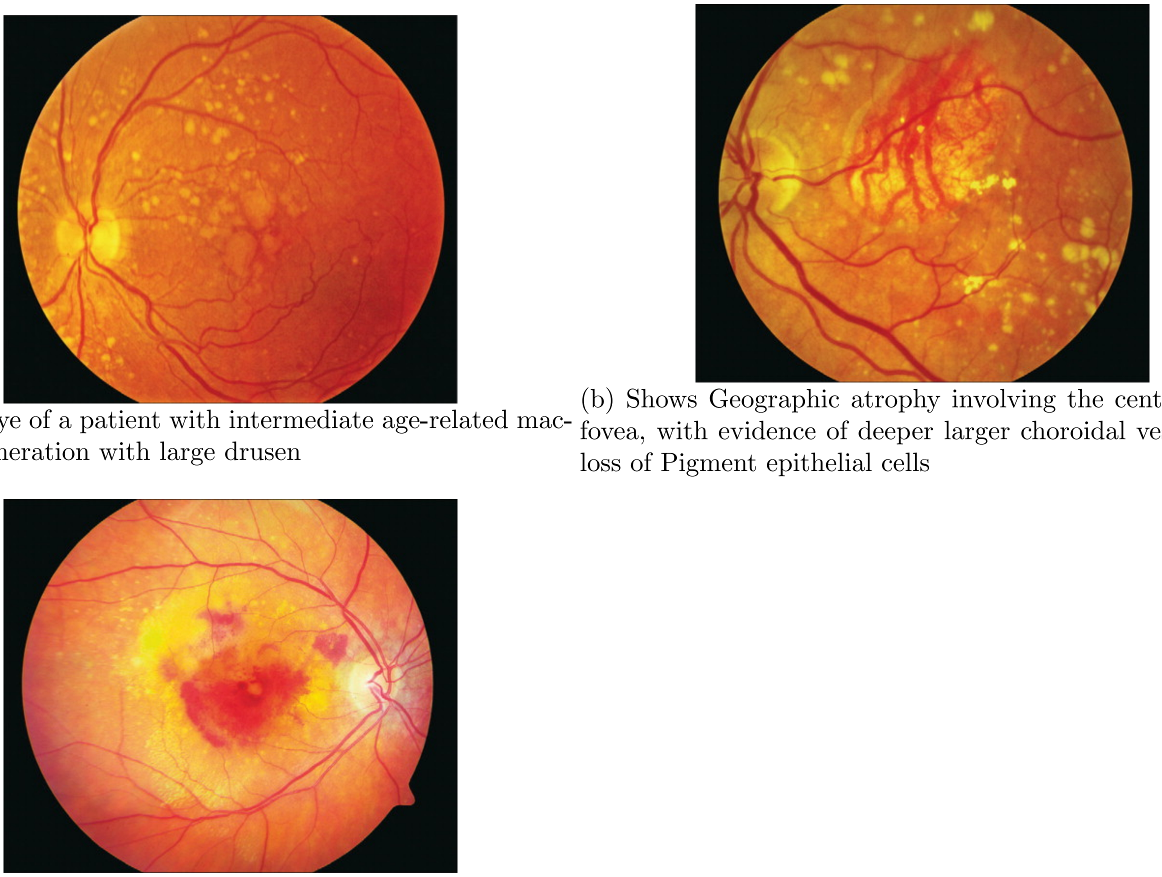
1.2.1 Classification of AMD

Early age-related macular degeneration
Multiple small drusen(s) (¡63 μm) or intermediate(s) ($\geq$ 63 μm and ¡125 μm) with no evidence of advanced age-related macular degeneration.

Intermediate age-related macular degeneration
Extensive intermediate drusen(s) or large drusen(s) ($\geq$125 μm) with no evidence of advanced age-related macular degeneration (figure 1).

Advanced age-related macular degeneration Advanced age-related macular degeneration is characterised by the presence of one or other of geographic atrophy or

Figure 1.2: Stages of Age-Related Macular Degeneration



(a) Left eye of a patient with intermediate age-related macular degeneration with large drusen

(b) Shows Geographic atrophy involving the centre of the fovea, with evidence of deeper larger choroidal vessels and loss of Pigment epithelial cells

(c) Neovascular age-related macular degeneration, with retinal haemorrhage, lipids, or retinal hard exudate and subretinal fluid neovascular age-related macular degeneration.

Geographic atrophy is the presence of a discrete area of retinal depigmentation at least 175 μm in diameter with a sharp border and visible choroidal vessels in the absence of neovascular age-related macular degeneration in the same eye (figure 2). 3 Geographic atrophy results from the continued loss of retinal pigment epithelium, with the eventual development of focal areas of total loss of the retina, retinal pigment epithelium, and the small blood vessels directly under the epithelium. The disease is generally slowly progressive. Central geographic atrophy involves the centre of the macular.

Neovascular age-related macular degeneration is characterised by the serous or hemorrhagic detachment of either the retinal pigment epithelium or sensory retina, the presence of sub retinal fibrous tissue, or minimal sub retinal fibrosis (figure 3). 3 Neovascularisation developing under the retina, which can leak fluid or bleed. Onset of vision loss is acute. Neovascular age-related macular degeneration is the most common cause of severe central visual loss. 4, 5 and 6

1.2.2 Scanning the Choroid

The Choroid layer of the eye is vital in the pathogenesis of many ocular conditions including age-related macular degeneration. This is because being able to scan the choroid in high resolution, could lead to being a valuable diagnostic tool for both the detection of monitoring of such conditions i.e. Age-Related Macular Diseases (see section 2.5).
Optical Coherence Tomography (OCT) allows non-invasive, *in-vivo* visualisation of the choroid. As a result, by using a series of techniques such as SS-OCT, we are able to view the structure of the eye, far more clearly than what was previously possible, despite its depth and the overlying *Retinal pigment epithelium* (RPE) layer. Other techniques that can be used for improved visualisation of the choroid are discussed later (see section **??** & section 2.6.5).

We will conclude the introduction and background section with the application of these topics in relation to my project with a discussion with regard to the hardware/software that was used during the completion of the project as well as potential applications of the project. In the ***Literature review*** chapter we will be discussing in depth: on some background information related to texture analysis and several applications of texture analysis in both related and unrelated fields. We will also discuss in a similar structure about Choroid diseases, alongside this there will be brief introduction to the Medical equipment that was used to gather the test data sample, namely, the Optical Coherence Tomography (OCT) Scanner.
Afterwards, the next sections will discuss the implementation of project, which will expand on the proposed solution in detail, discussing the principles methods, and a step-by-step

explanation on how the implementation of this experiment is carried out. Finally, a review is conducted on the results given by the implementation with remarks on possible improvements and potential advancement in this research, which will be discussed in the ***Conclusion*** Chapter of this report.

1.3 Project Specification

This section will briefly summarise the collected information in the introduction and Background sections, in order display the intended Aims & Objectives, The motivation of the project along with the potential impact of it, and finally a brief summary and explaining of required material for this project.

1.4 Aims and Objectives

- Aim: to create an automated classification system that would detect diseases that appear in the choroid layer of the eye.
- Aim: Test the feasibility of creating an automated classifier for choroid disease
- Aim: Identify any possible refinements/requirements to improve the classifier
- Objective: we will use a method called labelling which involves individually inspecting different medical scans of the choroid from a sample of patients and marking an area which the diseases or notable interest appears in the scan, the resultant of this is a region of interest (ROI), this provides the best quality of data samples, as each scan in treated and inspected manually, however as a result of this method, the process is labour intensive which means it is not practical to manually label large sets of data and there is the risk of human error during the labelling process.
- Objective: once a sufficient sample of data is labelled, we would experiment using several textures features selection methods which are mentioned previously in this report. In order to see what features provide the best results for classification, if however the experiment results are insufficient quality, other potential methods in either the labelling or classification process may be used in order to provide better results.
- Objective: in coalition with the second objective, a method of classification would be used, this is called a classifier, this is essentially a system that takes a sample of data, follows a set of rules/criteria which are formed previously from the texture feature analysis of the data, afterwards a process called the training process begins which the classifier repeats the process of understanding the data provided and identifying particular groups within the data set, with changes added and removed in order to

improve the accuracy of the classifier, the result of training the classifier allows the classifier to detect and classify diseases that have similar features to a particular groups of diseases that have been previously labelled. As a result, the detection and classification process for choroid diseases become more automated.

- Objective: conclude finding, determine the feasibility of the project, if not, determine what are the major hindrances of the projects for it to succeeds, if so, describe its application, what performs well and what could be improved, finally conclude on future directions to take from this project, describing potential improvements in the project and related fields of interest.

1.5 Motivation

This project was taken for several reasons, namely to develop basic knowledge in biomedical science (the fusion of technology in medicine), whilst also attempting to provide a solution to a current practical issue related to the labour intensive roles of classifying choroid diseases on a case by case basis, the resultant of this project could potentially provide a solution to this practical issue by automating the process making it more efficient whilst aiding in the decision making for diagnosis and treatment of mentioned choroid diseases. As this issue has yet to be resolved sufficiently, with a potential working and practical solution, this may lead to future developments in making automated classification for other forms of diseases.

1.6 Project Software & Hardware Requirements

As mentioned before in the background section, there are several key aspects of the project which need to addressed, namely the gathering of OCT images, processing the images, training the classifier (if possible), and validating the classifier. For these aspects to progress certain materials are required; Firstly the gathering of the OCT images will need to gathered using a OCT machine - this is provided by Cardiff University Hospital. Next software is required to process the images in order to be suitable for training. In this case a software package called ***Osirix***[Rosset et al. 2004], is used to display the OCT images, and allow the labelling of the images. Finally, a well known software environment known as ***MATLAB*** is used, to develop and run scripts that would perform three important parts of the projects; the processing of the labelled images, the training of the classifier and the validation of the classifier.

Author's Note: Due to the use of third party software namely **LibSVM** [C.-C. Chang and C.-J. Lin 2011] on both a Windows and Macintosh System, the additional requirement of a compiler (Visual Studio) is need to run the libraries provided by LibSVM on a Window System.

Chapter 2

Literature Review

This section will very light touch upon particular topic, in order to provide insight in several areas such as various texture analysis methods, information regarding choroid diseases and other information that may be interpreted as technical jargon.

2.1 Situation

Diseases within the human body has always existed throughout time, the eye is no exception to this rule, however with advances in medicine and technology, diseases in the eye has started to be identified and treatment have slowly been created to remedy these diseases. A particular example of diseases come in the form of Age-Related Muscular(ARM) diseases such as Age-related Muscular Degeneration (AMD), pigment epithelial detachments (PED) along with Neovascular AMD also known as Choroidal Neovascular Membranes (CNVM). As life expectancy increase due to various factors such as living style, and medical treatment, these forms of diseases are becoming a more common occurrence. Due to members of the public, living longer leading to bodily functions to eventually degenerate thus forming these diseases. An increase in these diseases combined with the specialist technology involved to scan the eye, means that more specialists are required in order to operate and diagnose these diseases, this leads to various problems such as increase in costs for both staff members and machine maintenance; shortage of specialist due to the training required; finally the increase in time required between scanning, identification and diagnosing the diseases, due to entire process being done manually.

2.2 Texture Analysis

2.2.1 Statistical Approaches

Statistical texture analysis methods measure the spatial distribution between pixel values. These methods are fundamental in the computer vision world and have been extensively applied to various tasks [Xie 2008]. Numerous statistical texture features have been proposed, ranging from first order statistics to higher order statistics. Along with many other various statistical methods such as: histogram statistics, co-occurrence matrices, auto correlation, and local binary patterns have been applied to visual inspection[Xie 2008].

2.2.2 Structural approaches

In structural approaches, the texture is characterised by texture primitives or texture elements, and the spatial arrangement of these primitives [Vilnrotter et al. 1986]. Thus, the primary goals of structural approaches is firstly to extract texture primitives, then model or generalise the spatial placement rules. The texture primitives can be as simple as individual pixels, a region with uniform grey levels, or line segments. The placement rules can be either obtained through modelling geometric relationships between primitives or learning the statistical properties of the placement rules via the texture primitives[Xie 2008].

2.2.3 Filter based approaches

Filter based approaches share a common characteristic, which is, applying filter banks on the image and computing the feedback given from the filter response. This approach can be divided into spatial domain, frequency domain, and joint spatial/spatial-frequency domain techniques.

2.2.4 Spatial and Frequency domain filtering

Measuring the strength and frequency of edges is one of the earliest attempts to discriminate different textures. In filtering in spatial domain, the images are usually filtered by gradient filters to extract edges, lines, isolated dots, etc. as a result several edge detection techniques such as Sobel, Robert, Canny, Laplacian, Deriche, Laws filters have been routinely used as a form of measuring edge density [Xie 2008]. Many other methods apply filtering in the frequency domain, particularly when no straightforward kernel can be found in the spatial domain [Xie 2008]. The image is transformed into the Fourier domain which involves a Fourier transform that decomposes a function of time into frequencies that form the composition of the function, it then multiplied with a filter function and the resultant is then re-transformed into the spatial domain as a result it becomes more efficient as it removes spatial convolution operation. Ring and wedge filters are some of the most commonly used frequency domain filters [Xie 2008].

2.2.5 Joint Spatial/Spatial-frequency methods

Since the Fourier coefficients is dependent on the entire image, the Fourier transform is not able to localise the defective regions in the spatial domain. The classical way of introducing spatial dependency into Fourier analysis is through a windowed Fourier transform. An example of this is if the window function is Gaussian, the windowed Fourier transform becomes the well-known Gabor transform, which can arguably achieve optimal localisation in the spatial and frequency domains. [Xie 2008]

2.2.6 Model Based Approaches

Model based methods include, among many others, fractal models, auto regressive models, random field models, the epitome model, and the texem model [Xie 2008]. An example of a model is the introduction of Fractals, that were initially proposed by Mandelbrot [Xie 2008], are geometric primitives that are self-similar and irregular in nature. Fragments of a fractal object are exact or statistical copies of the whole object and they can match the whole by stretching and shifting. Fractal dimension and lacunarity are both important measurements in fractal models, the former serves as a measure of complexity and irregularity; and the latter measures the structural variation or inhomogeneity [Xie 2008]

2.3 Classification

As mentioned, the aim of the textural analysis is detection and classification, this section covers the various approaches to a decision making scheme, these approaches can generally be divided into supervised learning and unsupervised (or semi-supervised) learning. In supervised learning the input pattern based on features derived from earlier stages, is identified as a member of a pre-defined known class. This approach has been widely used in visual inspection e.g. [A. Monadjemi n.d.; Kumar 2003; Wiltschi et al. 2000] A common and simple classifier is a technique called K-Nearest Neighbour (KNN) that uses supervised learning based on a distance algorithm where the pattern is assigned to the class shared by the majority of the K nearest neighbours. Another common approach that has been extensively used are, Artificial Neural Networks (ANN) for decision making procedures, this is due to ANN behaving with the ability to learn complex non-linear input-output relationships. Finally, there are *Support Vector machines (SVM)* that are used to classify surfaces. These machines are a form of supervised learning model that works with algorithms that analyse data and recognise patterns, which can be used in classification and regression analysis. In this report we will focus on supervised learning, as Unsupervised Learning methods such as *K Means*, do not require labels, it becomes a problem trying to find hidden structure in the unlabeled data. As there is no error or reward signal to evaluate a potential solution. Also, with regard to this project, as the experiment involve the use of ground truth labelling, it is more applicable to use a supervised learning method. That said, the project employed the use of an unsupervised learning algorithm known as a Deep Belief Network, this was added in the later stages of the project along with the use of a Restricted Boltzmann Machine, in order to produce a broader analysis on the choroidal data set, such that observations can be made regarding the possibility of using Deep Learning algorithm to create a more accurate classifier.

2.3.1 Machine Learning

Machine learning is a scientific principle that focuses on the construction and study of algorithms that can learn from [Kozak et al. 2015]Generally these algorithms operate by building a model from an example set of inputs, which as result utilise s this set to make predictions or decisions. Machine Learning is a sub-field off computer science, which shares it roots from research in artificial intelligence[Bishop 2006]. Machine learning typically applied to a range of computing asks where conventional design and implementation rule based algorithms that uses explicit rule is infeasible. Due to the complex patterns involves in the texture analysis, it becomes more beneficial to utilise machine learning in algorithms within classifiers in order to achieve some forms of classification.

2.3.2 K-Nearest Neighbour (k-NN)

In pattern recognition, the k-Nearest Neighbours algorithm (k-NN) is a non-parametric method used for classification and regression [Altman 1992]. In both cases, the input consists of the k closest training examples in the feature space and the output depends on whether k-NN is used for classification or regression:

In k-NN classification, the output is a class membership. This refers to an object being classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours (k is a positive integer, typically small). If (k = 1), then the object is simply assigned to the class of that single nearest neighbour. On the other hand in k-NN regression, the output is the property value for the object. This refers to the average of *k nearest neighbors* values. These neighbours are taken from a set of objects for which the class (for k-NN classification) or the object property value (for k-NN regression) is known. This can be thought of as the training set for the algorithm, though no explicit training step is required.

In classification and regression, it can be useful to assign weight(s) to the contributions of the neighbours, so that the nearer neighbours contribute more to the average rather than the more distant ones. For example, a common weighting scheme consists in giving each neighbour a weight of 1/d, where d is the distance to the neighbour.

Properties

K-NN is a special case of a variable-bandwidth, kernel density *balloon* estimator with a uniform kernel [Mills 2011; Terrell and Scott 1992]. The naive version of the algorithm is easy to implement by computing the distances from the test example to all stored examples, but it is computationally intensive for large training sets. The use of an appropriate nearest neighbour search algorithm in conjunction with k-NN make it more computationally tractable even for large data sets. In recent years there has been numerous nearest neighbour search algorithms proposed, which aim to reduce the number of distance evaluations actually performed making it more computationally efficient. Another property of k-NN, is that it has some strong consistency results. As the amount of data approaches infinity, the algorithm is guaranteed to yield an error rate no worse than twice the Bayes error rate (the minimum achievable error rate given the distribution of the data)[Cover and Hart 1967].

2.3.3 Random Decision Forest (RDF)

Random forests are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random forests, also correct the habit of overfitting to their training set that usually occurs in decision trees [see J. Friedman et al. 2001, p587-588].

The algorithm for inducing a random forest was developed by Leo Breiman [Breiman 2001] and Adele Cutler, [Liaw et al. 2009] and ”Random Forests” is their trademark. The method combines Breiman’s *bagging idea* and the random selection of features, introduced independently by Ho [Ho 1995, 1998] and Amit and Geman [Amit and Geman 1997] in order to construct a collection of decision trees with controlled variance.

Early development of random forests was influenced by the work of Amit and Geman [Amit and Geman 1997], who introduced the idea of searching a random subset of the available decisions whilst splitting a node, this is in the context of growing a single tree. The idea of random subspace selection came from Ho [Ho 1998], who played an influential part in the design of random forests. It is in this method that a forest of trees is grown, and variation(s) among the trees is introduced by via the insertion of training data that has undergone a random permutation in a randomly chosen subspace before fitting each tree. Finally, the idea of randomized node optimization, where the decision at each node is selected by a randomized procedure, rather than a deterministic optimization was first introduced by Dietterich [Dietterich 2000].

This led to the introduction of a proper random forest was made in a paper by Leo Breiman.[Breiman 2001] This paper describes a method of building a forest of uncorrelated trees using a CART like procedure, combined with randomized node optimization and bagging. In addition, this paper combines several ingredients, some previously known and some novel, which form the basis of the modern practice of random forests, in particular: the use out-of-bag error as an estimate of the generalization error. Also, the Measurement of a variables importance through permutation.

Properties

Variable importance

Random forests can be used to rank the importance of variables in a regression or classification problem naturally. The following technique was described in Breiman's original paper[Breiman 2001] and is implemented in the R package randomForest.[Liaw et al. 2009]

The first step in measuring the variable importance in a data set $\mathcal{D}_n = \{(X_i, Y_i)\}_{i=1}^n$ is to fit a random forest to the data. During the fitting process the out-of-bag error for each data point is recorded and averaged over the forest (errors on an independent test set can be substituted if bagging is not used during training).

To measure the importance of the j-th feature after training, the values of the j-th feature are permuted among the training data and the out-of-bag error is again computed on this perturbed data set. The importance score for the j-th feature is computed by averaging the difference in out-of-bag error before and after the permutation over all trees. The score is normalized by the standard deviation of these differences, this result in, features which produce large values being ranked as more important than features which produce small values.

This method of determining variable importance has some drawbacks. For data including categorical variables with different number of levels, random forests are biased in favor of those attributes with more levels. Methods such as partial permutations[Altmann et al. 2010; H. Deng et al. 2011][12] and growing unbiased trees[13] can be used to solve the problem.

Relationship to nearest neighbors

A relationship between random forests and the k-nearest neighbour algorithm (k-NN) was pointed out by Lin and Jeon in 2002.[Y. Lin and Jeon 2006] It turns out that both can be viewed as so-called weighted neighborhoods schemes. These are models built from a training set $\{(x_i, y_i)\}_{i=1}^n$ that make predictions $\hat{y}$ for new points x$'$ by looking at the *neighborhood* of the point, formalized by a weight function W:

$$\hat{y} = \sum_{i=1}^{n} W(x_i, x')\, y_i. \tag{2.1}$$

Where, $W(x_i, x')$ is the non-negative weight of the i'th training point relative to the new point x'. For any particular x', the weights must sum to one. Weight functions are given as follows:

- In k-NN, the weights are $W(x_i, x') = \frac{1}{k}$ if xi is one of the k points closest to x', and zero otherwise.
- In a tree, $W(x_i, x') = \frac{1}{k'}$ if x_i is one of the k' points in the same leaf as x', and zero otherwise.

Since a forest averages the predictions of a set of m trees with individual weight functions W_j, its predictions are

$$\hat{y} = \frac{1}{m}\sum_{j=1}^{m}\sum_{i=1}^{n} W_j(x_i, x')\, y_i = \sum_{i=1}^{n}\left(\frac{1}{m}\sum_{j=1}^{m} W_j(x_i, x')\right) y_i. \tag{2.2}$$

This shows that the whole forest is again a weighted neighborhood scheme, with weights that average those of the individual trees. The neighbours of x$'$ in this interpretation are the points x_i which fall in the same leaf as x$'$ in at least one tree of the forest. In this way, the neighbourhood of x$'$ depends on the complexities of the tree structure, and thus on the structure of the training set. Lin and Jeon show that the shape of the neighbourhood used by a random forest adapts to the local importance of each feature [Y. Lin and Jeon 2006].

2.3.4 Support Vector Machines

In machine learning, support vector machines (SVMs, also support vector networks[1]) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. Given a set of training examples, each marked for belonging to one of two categories, an SVM training algorithm builds a model that assigns new examples into one category or the other, making it a non-probabilistic binary linear classifier. An SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted to belong to a category based on which side of the gap they fall on. In addition to performing linear classification, SVMs can efficiently perform a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

The first SVM algorithm was invented by Vladimir N. Vapnik and Alexey Ya. Chervonenkis in 1963. An adaptation was made by Bernhard E. Boser, Isabelle M. Guyon and Vladimir N. Vapnik in 1992 that suggested a way to create nonlinear classifiers by applying the kernel trick to maximum-margin hyperplanes [Boser et al. 1992]. The current standard incarnation (soft margin) was proposed by [Cortes and Vapnik 1995] A Support Vector Machines works by first be given a training dataset, with data examples marked as belonging to one of two categories, an SVM training algorithm would be applied to build models that assign new data examples into one category or the other, this type of classification is called a non-probabilistic binary linear classifier.
A Support Vector Machine visualises the resultant model as points in space that have been mapped to represent groups that are separated by a divider called the functional margin. It is possible to then perform further adjustments to parameters and variables in the SVM, in order to increase the dividing gap between groups. Resulting, in an improvement in the overall result, as there are more clearly defined groups making classification for new entries more accurate.

Motivation

Classifying data is a common task in machine learning. Suppose some given data points each belong to one of two classes, and the goal is to decide which class a new data point will be in. In the case of support vector machines, given a data point that is viewed as a **p**-dimensional vector (a list of p numbers), the possibility of separating such points with a $(\mathbf{p}-\mathbf{1})$-dimensional hyperplane is known, as a linear classifier.
There are numerous hyperplanes that could be used to classify the data, one can assume the most appropriate hyperplane would be one that represents the largest separation, or margin, between the two classes.this is achieved by choosing the hyperplane, such that the distance from it to the nearest data point on each side is maximized. If such a hyperplane exists, it is known as the maximum-margin hyperplane and the linear classifier it defines is known as a maximum margin classifier; or equivalently, the *perceptron of optimal stability.*

Properties

SVMs belong to a family of generalized linear classifiers and can be interpreted as an extension of the perceptron. They can also be considered a special case of Tikhonov regularization. A special property is that they simultaneously minimize the empirical classification error and maximize the geometric margin; hence they are also known as *maximum margin classifiers.*

Parameter selection

The effectiveness of SVM depends on the selection of kernel, the kernel's parameters, and soft margin parameter C. A usual choice is a Gaussian kernel, which has a single parameter γ. The best combination of C and γ is often selected by a grid search with exponentially growing sequences of C and γ, for example,$C \in \{2^{-5}, 2^{-3}, \ldots, 2^{13}, 2^{15}\}; \gamma \in$

$\{2^{-15}, 2^{-13}, \ldots, 2^{1}, 2^{3}\}$. Typically, each combination of parameter choices is checked using cross validation, and the parameters with the best cross-validation accuracy are picked. Alternatively, recent work in Bayesian optimization can be used to select C and γ , often requiring the evaluation of far fewer parameter combinations than grid search. The final model, which is used for testing and for classifying new data, is then trained on the whole training set using the selected parameters [Hsu et al. 2003]. On the other hand SVM has potential drawbacks for example: Uncalibrated class membership probabilities, SVM is only directly applicable for two-class tasks. Therefore, algorithms that reduce the multi-class task to several binary problems have to be applied and finally parameters of a solved model can be difficult to interpret.

2.4 Deep Learning

Deep learning (Deep Machine Learning, or Deep Structured Learning, or Hierarchical Learning, or sometimes DL) is a branch of machine learning based on a set of algorithms that attempt to model high-level abstractions in data by using model architectures, with complex structures or otherwise, composed of multiple non-linear transformations [Bengio 2009; Bengio, Courville, et al. 2013; L. Deng and Yu 2014; Schmidhuber 2015].

Deep learning is part of a broader family of machine learning methods based on learning representations of data. An observation (e.g., an image) can be represented in many ways such as a vector of intensity values per pixel, or in a more abstract way as a set of edges, regions of particular shape etc... Some representations make it easier to learn tasks i.e. face recognition or facial expression recognition [Glauner 2015] from examples. One of the promises of deep learning is replacing hand crafted features with efficient algorithms for unsupervised or semi-supervised feature learning and hierarchical feature extraction [Song and Lee 2013].

Various deep learning architectures such as *deep neural networks (DNN)*, *convolutional deep neural networks (CNN)*, *deep belief networks (DBN)* and *recurrent neural networks (RNN)* have been applied to fields like computer vision, automatic speech recognition, natural language processing, audio recognition and bioinformatics where they have been shown to produce state-of-the-art results on various tasks.

2.4.1 Restricted Boltzmann Machines (RBM)

A Restricted Boltzmann Machine (RBM) is a generative stochastic artificial neural network that can learn a probability distribution over its set of inputs.
RBMs were initially invented under the name Harmonium by Paul Smolensky in 1986,[Smolensky 1986] but only rose to prominence after Geoffrey Hinton and collaborators invented fast learning algorithms for them in the mid-2000s. RBMs have found

applications in dimensionality reduction, [G. E. Hinton and R. R. Salakhutdinov 2006] classification, [Larochelle and Bengio 2008] collaborative filtering, [R. Salakhutdinov et al. 2007] feature learning[Coates et al. 2011] and topic modelling.[G. E. Hinton and R. R. Salakhutdinov 2009] They can be trained in either supervised or unsupervised ways, depending on the task.

As their name implies, RBMs are a variant of Boltzmann machines, with the restriction that their neurons must form what is known as a bipartite graph, which is where a pair of nodes from each of the two groups of units, commonly referred to as the visible and hidden units respectively, may have a symmetric connection between them, and have no connections between each other within a group. By contrast, unrestricted Boltzmann machines may have connections between hidden units. This restriction allows for more efficient training algorithms than are available for the general class of Boltzmann machines, in particular the gradient-based contrastive divergence algorithm.[Carreira-Perpinan and G. E. Hinton 2005]

Structure

The standard type of RBM has binary-valued (*Boolean/Bernoulli*) hidden and visible units, and consists of a matrix of weights W = ($w_{i,j}$) (size mn) associated with the connection between hidden unit h_j and visible unit v_i, as well as bias weights (offsets) a_i for the visible units and b_j for the hidden units. Given these, the energy of a configuration (pair of Boolean vectors) (v,h) is defined as

$$E(v,h) = -\sum_i a_i v_i - \sum_j b_j h_j - \sum_i \sum_j v_i w_{i,j} h_j$$

or, in matrix notation,

$$E(v,h) = -a^{\mathrm{T}} v - b^{\mathrm{T}} h - v^{\mathrm{T}} W h \tag{2.3}$$

This energy function is analogous to that of a Hopfield network. As in general Boltzmann machines, probability distributions over hidden and/or visible vectors are defined in terms of the energy function:[G. Hinton 2010]

$$P(v,h) = \frac{1}{Z} e^{-E(v,h)} \tag{2.4}$$

Where Z is a partition function defined as the sum of $e^{-E(v,h)}$ over all possible configurations (in other words, just a normalizing constant to ensure the probability distribution sums to 1). Similarly, the (marginal) probability of a visible (input) vector of Boolean(s) is the sum over all possible hidden layer configurations:[G. Hinton 2010]

$$P(v) = \frac{1}{Z} \sum_h e^{-E(v,h)} \tag{2.5}$$

Since the RBM follows the shape of a bipartite graph, with no intra-layer connections, the hidden unit activations are mutually independent given the visible unit activations and conversely, the visible unit activations are mutually independent given the hidden unit activations [Carreira-Perpinan and G. E. Hinton 2005]. Therefore, for m visible units and n hidden units, the conditional probability of a configuration of the visible units v, given a configuration of the hidden units h, is:

$$P(v|h) = \prod_{i=1}^{m} P(v_i|h) \tag{2.6}$$

Conversely, the conditional probability of h given v is

$$P(h|v) = \prod_{j=1}^{n} P(h_j|v). \tag{2.7}$$

The individual activation probabilities are given by

$$P(h_j = 1|v) = \sigma\left(b_j + \sum_{i=1}^{m} w_{i,j} v_i\right) \, and \, P(v_i = 1|h) = \sigma\left(a_i + \sum_{j=1}^{n} w_{i,j} h_j\right) \tag{2.8}$$

where σ denotes the logistic sigmoid.

The visible units of RBM can be multinomial, although the hidden units are Bernoulli. In this case, the logistic function for visible units is replaced by the Softmax function

$$P(v_i^k = 1|h) = \frac{\exp(a_i^k + \Sigma_j W_{ij}^k h_j)}{\Sigma_{k=1}^K \exp(a_i^k + \Sigma_j W_{ij}^k h_j)} \tag{2.9}$$

where K is the number of discrete values that the visible values have.

2.4.2 Deep Belief Network (DBN)

In machine learning, a deep belief network (DBN) is a generative graphical model, or a type of deep neural network, composed of multiple layers of latent variables (hidden units), with connections between the layers but not between units within each layer.[G. E. Hinton 2009]

When trained on a set of examples without aid, a DBN can learn to probabilistically reconstruct its inputs. The layers then act as feature detectors on inputs.[G. E. Hinton 2009] After this learning step, a DBN can be further trained in a supervised way to perform classification.[G. E. Hinton and R. R. Salakhutdinov 2006]

DBNs can be viewed as a composition of simple, unsupervised networks such as restricted Boltzmann machines (RBMs)[G. E. Hinton 2009] or auto-encoders,[Bengio, Lamblin, et al. 2007] where each sub-network's hidden layer serves as the visible layer for the next. This also leads to a fast, layer-by-layer unsupervised training procedure, where contrastive divergence is applied to each sub-network in turn, starting from the ”lowest” pair of layers (the lowest visible layer being a training set).

Training Algorithm
The training algorithm for DBNs proceeds as follows.[G. E. Hinton and R. R. Salakhutdinov 2006] Let X be a matrix of inputs, regarded as a set of feature vectors.

- Train a restricted Boltzmann machines on X to obtain its weight matrix, W. Use this as the weight matrix for between the lower two layers of the network.
- Transform X by the RBM to produce new data X', either by sampling or by computing the mean activation of the hidden units.
- Repeat this procedure with X X' for the next pair of layers, until the top two layers of the network are reached.

2.5 Choroidal Diseases and Disorders

The term choroid is derived from the Greek words for *membrane* and *form.* Histological investigations of this tissue have been performed since the 17th century [Regatieri et al. 2012]. The choroid is a vascular and pigmented membrane layer that extends from the *ora serrata* to the optic nerve and lies between the retina layer and the sclera layer (see figure 1.1 for reference)

Age-related macular degeneration (AMD) is the most common cause of irreversible blindness, it can be categorised into two types: dry (non-exudative or non-neovascular) and wet (exudative or neovascular)[Spraul et al. 1999] (see figure 2.1 for reference). Optical coherence tomography, together with other medical scanning techniques such as, clinical bio-microscopy, colour fundus photography, auto fluorescence imaging etc. are used to evaluate patients with AMD. Scans called Macular thickness maps are generated by the OCT and have been shown to be useful in monitoring the progression and treatment response to disease states, such as neovascular AMD after antiangiogenic treatment[Regatieri et al. 2012]. The choroidal structure in AMD is of particular interest as it been hypothesized that abnormalities within the choroidal circulation, may contribute to the development of AMD [Regatieri et al. 2012]. In a report by Spaide [Regatieri et al. 2012], it concludes that choroidal thickness decreases with increasing age and the choroidal circulation might play a role in the pathophysiology of AMD. In another study sub foveal choroidal thickness in eyes with neovascular AMD were compared with eye with polypoidal choroidal vasculopathy (PCV), it showed that eyes with PCV have a thicker sub foveal choroid [around 293 +/-72.3] compared to eyes with neovascular AMD (around 245 +/- 73.1)[Koizumi et al.

2011], Additionally, it was found that eyes with a sub foveal choroidal thickness of ≥ 300 are more than 5 times more likely to have PCV. Similar results have been reports in different experiments.[Kim et al. 2011; Koizumi et al. 2011], all these results indicate that the thickness of the choroid may play a role in the differentiating between different types of diseases as indicated previously in [Koizumi et al. 2011] where the choroid is thinner in eyes with neovascular AMD and the choroid is thicker in eyes with PCV.

2.5.1 Age Related Macular Degeneration (AMD)

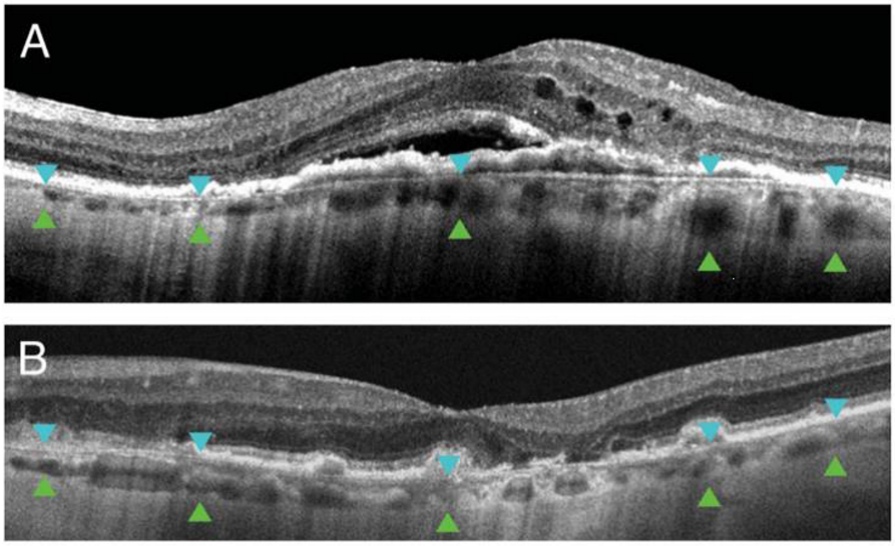


Figure 2.1: A shows an OCT image of a choroid with wet (exudative) AMD. B shows an OCT image of a choroid with dry AMD

2.5.2 Causes

Although the exact cause of AMD is not yet known studies have shown see Coleman et al. 2008, Risk Factors, it shows that age is the strongest cause for age-related macular degeneration, studies have shown that the prevalence of age-related macular degeneration increases with age in white individuals [Bird et al. 1995; N. Bressler and S. Bressler 1989; Cruickshanks 1997; D. Friedman et al. 1999; R. Klein, B. Klein, et al. 1992; Mitchell et al. 1995; VanNewkirk et al. 2000]. In [Coleman et al. 2008] its shown that gender is a possible cause, with women in particular having a risk as twice as much as men in individual aged over 75 [Smith, Assink, et al. 2001]. Results from controlled studies[Delcourt 1998; A.-R. E. D. S. R. Group 2000; E. D. C.-C. S. Group 1992; Hyman and Lilienfeld 1983; McCarty and Mukesh 2001; Vinding et al. 1992] and population-based studies [R. Klein, B. E. Klein, Linton, et al. 1993; Smith, Mitchell, and Leeder 1996; Vingerling 1996]and prospective studies of both physicians[Seddon, Willett, et al. 1996] and nurses[Christen et al. 1996] strongly confirm cigarette smoking as a risk factor for age-related macular degeneration. There exists a direct association between the risk of developing advanced age-related macular degeneration with the number of cigarettes smoked[Delcourt 1998; Khan et al. 2006].

Increasing evidence has shown that there are genetic factors involved in age-related macular degeneration.[Haddad et al. 2006; Scholl et al. 2007] Results from population-based studies implicate a familial component in the disease's pathogenesis, [Heiba et al. 1994; Klaver et al. 1998; Seddon, Ajani, et al. 1997; Smith and Mitchell 1998] as have studies of monozygotic[Gottfredsdottir et al. 1999; M. L. Klein et al. 1994] and dizygotic[Meyers et al. 1995] twins.[Seddon, Cote, et al. 2005] Klaver and colleagues[Klaver et al. 1998] have estimated that siblings of individuals with end-stage age-related macular degeneration have around five-fold increased risk for intermediate age-related macular degeneration and a twenty-fold increased risk for advanced age-related macular degeneration. They suggest that about 23% of advanced age-related macular degeneration can be attributed to a genetic component.

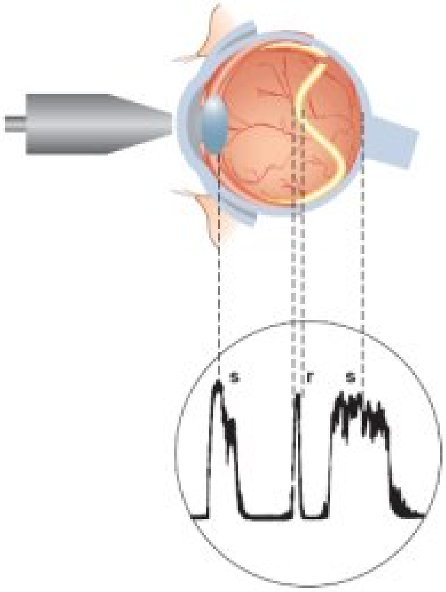
It has also been shown in a Multi-ethnic Study of Atherosclerosis that the prevalence of age-related macular degeneration in individuals aged 4585 years to be 24% in African-American people, 42% in Hispanic individuals, 46% in Chinese, and 54% in white individuals [R. Klein, B. E. Klein, Knudtson, et al. 2006]. It has been suggested that neovascular form of age-related macular degeneration was highest in Chinese individuals, compared with white people [Vinding et al. 1992]. Though these ethnic differences in prevalence need further validation[Coleman et al. 2008]. Finally, some report have suggested that an individuals dietary can affect the risk of age-related macular degeneration, this however has shown to be inconsistent, the Rotterdam Study, a high dietary intake of beta-carotene, vitamins C and E, and zinc was associated with a substantially reduced risk of age-related macular degeneration in elderly individuals.[Leeuwen et al. 2005] on the other hand, a trial of vitamin E in an Australian population with lower risk of developing advanced age-related macular degeneration failed to find a beneficial effect.[Taylor et al. 2002]

2.6 Optical Coherence Tomography (OCT)

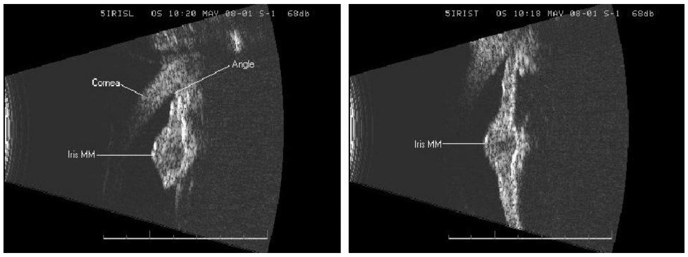
Optical coherence tomography (OCT) is an optical imaging modality, allowing cross-sectional visualisation of biological tissue within living organism. one of its advantages is that, it is non-invasive, and provides a high resolution view of internal structures, making it ideal for medical diagnostic purposes. Namely, in Ophthalmic diagnosis(the diagnosis pertaining to the eye) which is one of the most clinically developed applications for OCT.

Optical Coherence Tomography (OCT) has been used in the diagnosis of several eye conditions such as, Macular edema Age-Related Macular Degeneration and Central serous retinopathy to name a few examples, In addition, OCT is often used to evaluate disorders that are shown in the optic nerve. This id done by examining the optic nerve, which is made up of many nerve fibres and sends signals from your retina to your brain. These fibres can be viewed via OCT for any changes in its structure, in order to determine if the change is caused by conditions such as glaucoma [Kozak et al. 2015].
Due to OCT's dependency on a solid light source, it becomes more difficult to localise defects within any condition that interferes with light passing through the eye, such as dense cataracts or significant bleeding in the vitreous. [Kozak et al. 2015]

2.6.1 History of OCT



(a) Example of A scan


5IRISL OS 10:20 MAY 08-01 S-1 68db
Angle
Cornea
Iris MM
5IRIST OS 10:18 MAY 08-01 S-1 68db
Iris MM

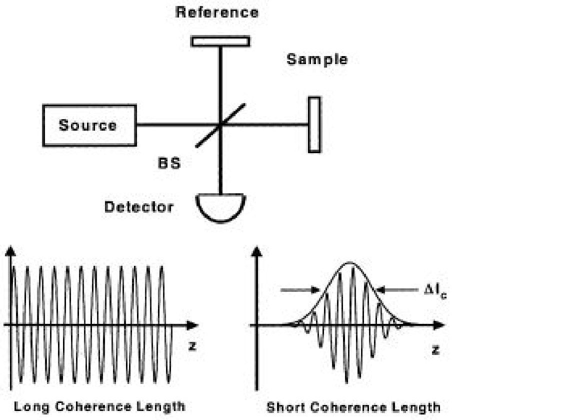
(b) Example of B scan

The first early recorded account of single dimensional axial scan(equivalent to an ultrasound A scan see fig 2.2a), were developed and used in the 1980s [Fercher, Mengedoht, et al. 1988; Fujimoto et al. 1986], the use of cross-sectional two-dimensional scans (equivalent to an ultrasound B scan see fig 2.2b) being shown in practice appeared in the early 1990s [Huang et al. 1991]. However the first actual OCT imaging scans within a living organism were recorded in studies of the human retina that appeared in 1993 [Fercher, Hitzenberger, et al. 1993; Swanson et al. 1993]. Since then, there has been a rapid and significant growth in direct use and related development in this technology.

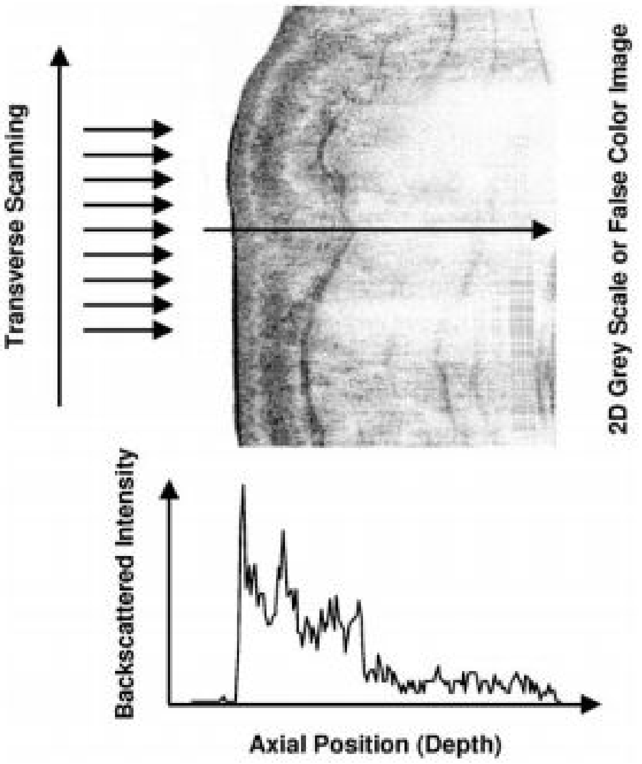
It has reached the point where commercial OCT instruments are now commonplace in hospitals, medical imaging centres and high street optometry practises.

2.6.2 Principles

it is said by [Fujimoto 2001] that Optical Coherence Tomography comparable to the principles used in ultrasound with measurement and operation being done with light instead. by measuring the level of reflected light from within the targeted area, whilst measuring the delay is sending and receive light, its possible to calculate from the time delay,to form cross sectional scans. However due to the extremely high velocity of light, it is not possible to measure the time delay [Fujimoto 2001]. instead one technique that was used to measure this time delay is through the use of low-coherence interferometry. which was first developed for measuring reflections in fiber optics and optoelectronic devices [Fujimoto 2001]. this technique is used to measure the echo time delay and intensity of back scattered light by interfering it with light that has traveled a known reference path length and time delay[Fujimoto 2001].



(a) OCT measures the echo time delay of reflected light by using low-coherence interferometry. The system is based on a Michelson-type interferometer (see figure 2.4). Reflections or backscattering from the object being imaged are correlated with light which travels a reference path [Fujimoto 2001].



(b) Cross-sectional images are constructed by performing measurements of the echo time delay of light at different transverse positions. The result is a two-dimensional data set that represents the backscattering in a cross-sectional plane of the tissue [Fujimoto 2001].

In essence by measuring at regular intervals in a given area via a Michelson-type interferometer (see figure 2.3a) . in extension to this by performing several scans in a traversed direction, the result is the production of a two dimensional profile which is referred as a *b-scan* (see figure 2.3b, In addition to lateral movement in one direction producing two-dimensional images though the movement in two directions it is plausible to create a more in detail profile, resulting in a three-dimensional image called a *c-scan* which are comparable to scans seen in modern medicine via the use of machines such as MRI

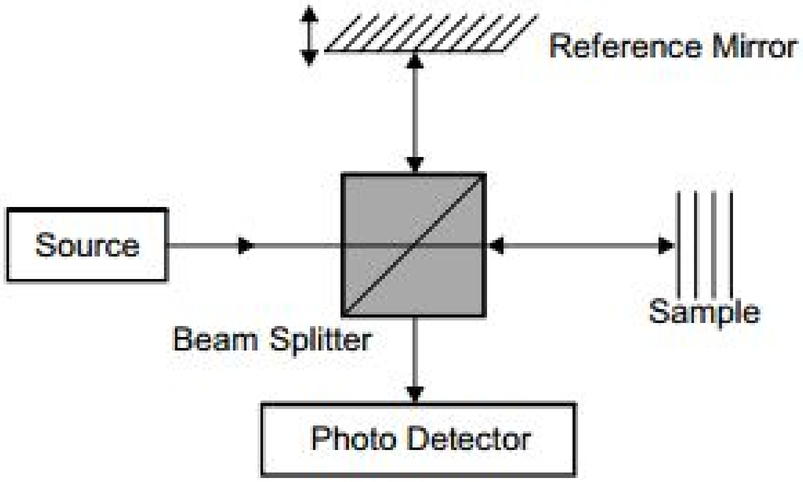


Figure 2.4: Schematic of OCT System Based on Michelson Interferometer [Ali and Parlapalli 2010]

An interferometer in an OCT instrument takes the light from a source is directed onto a beam splitter here, one of the beams is incident onto the sample to be imaged, while the second beam travels on a reference path with a variable path length[Fujimoto 2001] by taking the pulse that enters the eye i.e the first beam, and a reference pulse i.e the second beam, a small portion of the light is reflected by each retinal layer is interfered with reflected light from the reference arm and detected with a photo detector at the interferometer output [Fujimoto 2001]. Through the adjustment of these pulse of light, if they are adjust to a coherent length with each other, the resultant is the observation of interference fringes that are the result of the varying reflectivity levels of each retina layer.

The echo time delay and the intensity of the back scattered light from within the sample area can be measured by detecting, extracting and comparing the interference output of the interferometer whilst scanning the reference path length. This information enables the construction of an image that is of several depth points within the retina often refered as a(*a-scan*). However, with the occurrence of the scattering and absorption light this result in the OCT penetration depth in most human tissues to be limited to a few millimeters [Fercher 2010; Huang et al. 1991].

Optical coherence tomography comes in two flavours, commercial OCT which is more economical in its light sources producing great resolutions, however as a result there is reduced depth perception due to the strength of the light-source. The other kind are long-wave OCT which has a greater light source provides benefits such as reduced intraocular scattering; reduced absorption due to ocular pigmentation(melanin) and improved penetration of skin (cataract). The disadvantage that comes with this variation of OCT

Scanner, is a less focused source of light compared to the commercial OCT variant, resulting in a reduced axial resolution which affects the overall quality of the OCT scans[Kozak et al. 2015]. (further discussion in resolution is discussed below)

2.6.3 Resolution

In comparision to conventional microscopy, the inner working that control the axial and transverse image resolutions are independent from each other [Fujimoto 2001], in actuality the axial resolution in OCT is determined by the coherence length of the light source [Fujimoto 2001] , as a result higher resolutions can be obtained regardless of the focusing conditions of the beam [Fujimoto 2001]. **note:** the coherence length is obtained by calculating the spatial width of a field's auto-correlation [Fujimoto 2001]. in addition , this field auto correlation is comparable to the Fourier transformation of the light spectrum. meaning the width of the autocorrelation function or axial resolution is inversely proportional to the width of the power spectrum (see figure 2.5) and figure 2.3a. the result is the axial resolution can be calculated with the following equation

$$\Delta z = \frac{2\ln(2)\lambda^2}{\pi\Delta\lambda} \tag{2.10}$$

Where:
Δz, is the axial resolutiom / autocorrelation function,
$\Delta\lambda$, is the power spectrum and
λ, is the centre wavelength of the light source used for imaging (this assumes the use of a Gaussian spectrum).

in essence as the result of the axial resolution being inversely proportional to bandwidth of the light source, by extending the bandwidth of the light source, a higher resolution is achieved [Drexler 2004; Fujimoto 2001]. which has led to the development of *Ultrahigh-Resolution Optical Coherence Tomography* (UHR-OCT) with an axial resolution of 2-3μm according to [Drexler 2004; Drexler and Fujimoto 2008]. possible use cases of this technology is the scanning of living tissue, at a much more accurate and minute scale compared to before, the level of detail and accuracy is comparable to post-mortem histology, and has been used to visualise photoreceptor layer impairment in macular pathologies including *AMD, macular holes, Stargardts dystrophy*, and *retinitis pigmentosa* as shown in cases done by [Drexler, Sattmann, et al. 2003; Ergun et al. 2005; Ko et al. 2005; Witkin et al. 2006].

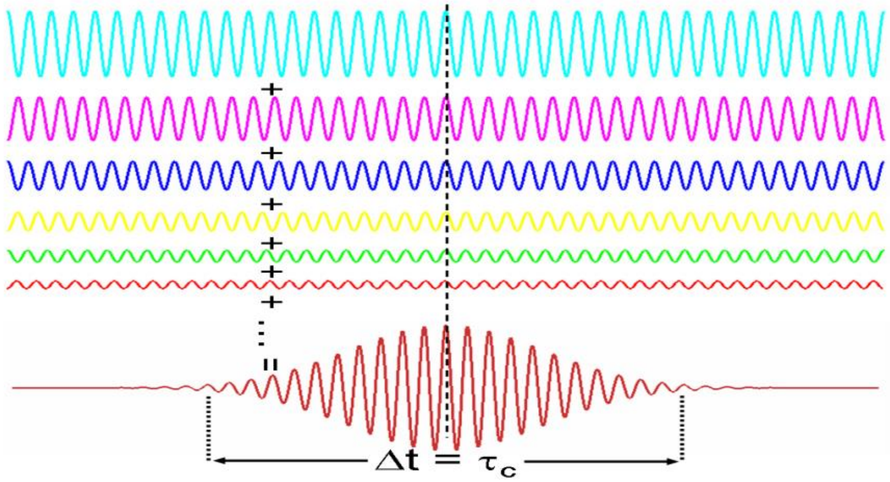


Figure 2.5: Diagram showing a sample of the coherence length (Δt)
Source:`https://en.wikipedia.org/wiki/Coherence_(physics)`

Transverse Resolution

The other form of resolution is the Traverse Resolution, this is determined by the following equation

$$\Delta x = \left(\frac{4\lambda}{\pi}\right)\left(\frac{f}{d}\right) \tag{2.11}$$

Where:
Δx, is the transverse resolution,
d, is the spot size on the objective lens
f, is its focal length
λ, is again the centre wavelength of the light source

according to [Drexler 2004; Fujimoto 2001] its posible to obtain high transverse resolution by using a large aperture and focusing the beams to a small spot size. In addition, [Fujimoto 2001] indicates the transverse resolution is also related to the depth of focus or the confocal parameter b which is 2zR, two times the Raleigh range and as such can be interpreted as the following equation

$$2\text{z}R = \frac{\pi \Delta x^2}{2\lambda} \tag{2.12}$$

However [Fujimoto 2001] noted that improving the transverse resolution led to a decrease in the depth of focus, similar to conventional microscopy. Other factors that can comprimise the transverse resolution is the condition of the media, for example Transverse image resolution of the retina is typically limited by pathological or normal ageing changes to the *intraocular* media (i.e. cataracts) and ocular peculiarities. but development in technology has been made to reduce these effect through the use of adaptive optics.

2.6.4 Types of OCT

There are several types of OCT which are catorized by a domain type. there is two main branch of Optical Coherence Tomography *Time Domain (TD-OCT) and Spectral Domain (SD-OCT)* [Drexler and Fujimoto 2008 with *Swept Source (SS-OCT)* and Fourier Domain (FD-OCT) coming as the product of development in OCT.

Time Domain OCT (TD-OCT)

in OCT by providing the path length to the reference arm and is aligned to ensure the tissue match to within the coherence length of the light source, when the reflected beams recombine, interference occurs, which is the intensity information gathered [Gabriele et al. 2011].By making manual adjustments to the location of the reference mirror, this allows back-scattered tissue intensity levels to be detected from different depths in the tissue sample. This approach is referred to as time-domain (TD)-OCT because time-encoded signals are obtained directly [Gabriele et al. 2011]. despite there being improvements in the TD-OCT [Gabriele et al. 2011, its fundamental limitation was the need for manual movement within the system. According to [Drexler 2004; Považay et al. 2007], this limitation meant that capturing images was a slow process, hence it is difficult to collect enough data to allow for UHR or three-dimensional imaging. As a result iterations of the TD-OCT such as the Stratus OCT (Carl Zeiss) system, are now considered obsolete in modern medicine [Drexler 2004].

in [Gabriele et al. 2011] report, the implementation of broadband light sources into OCT [Drexler, Morgner, et al. 1999] systems has improved the axial resolution from 10 m to as high as 2 m in tissue. in addition acquisition speed has improved considerably by detecting back-scattering signals in the frequency domain, which means back-scattered depth information at a given location can be collected without the movement of a reference mirror [Gabriele et al. 2011],and is converted from interferometry data [Velthoven et al. 2007]. the technology is used in two forms, Frequency information is acquired with a broad-bandwidth light source, charge-coupled device (CCD) camera, and a spectrometer or by sweeping a narrow-bandwidth source through a broad range of frequencies with a photo-detector [Gabriele et al. 2011]. These are referred to as Spectral Domain OCT (SD-OCT) and Swept Source OCT (SS-OCT) respectively further more accordining to [Gabriele et al. 2011 in both approaches, intensity profiles (A-scans) are obtained using a Fourier-transform of the detected frequencies, which allows a rapid A-scan collection. In addition to improved scanning speed, frequency-domain-OCT also offers the advantage of higher detection sensitivitythat is, it exhibits higher signal-to-noise, given a perfect reflector[Gabriele et al. 2011], further information on these two types of OCT are mentioned below.

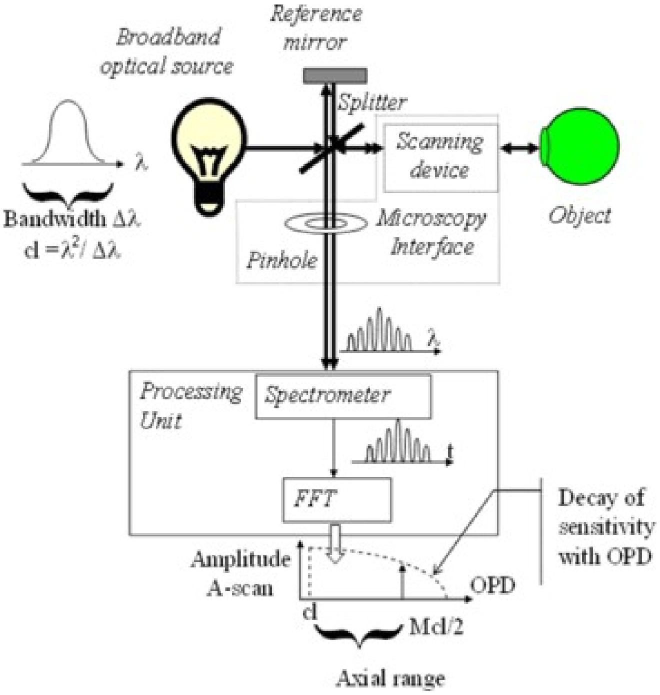


Figure 2.6: Diagram showing a typical SD-OCT setup [Podoleanu 2014]

Spectral Domain OCT (SD-OCT)

This is one of the earliest type of OCT to use a broadband light source with a *dispersive spectrometer* in the detector. SD-OCT refers to spectral interrogation of the spectrum at the interferometer output [Podoleanu 2014].In Figure 2.6, a broadband optical source is used and the Processing Unit uses a spectrometer, usually built using a prism or a diffraction grating and a linear photodetector array [Podoleanu 2014], this is used to distribute different wavelengths of light in a fashion similar to a spectrum. *Spectral Information* is extracted from this spectrum by detecting the various wavelengths via the processing unit via the use of a detector stripe. This results in the processing of full depth OCT image, using a single point of exposure of light from its source. As there is further development in this technology, an alternative approach is used called **SB-OCT**. This method is an extension of the work on white light interferometry with initial applications in absolute ranging and sensing [Podoleanu 2014].Similarly to SD-OCT, SB-OCT operates based on the optical spectrum output extracted from data provided by the low coherence interferometer[Podoleanu 2014]. Due to its sensitivity advantage and availability of fast digital linear cameras, the SB-OCT became the method of choice in current OCT investigations of the retina with video-rate images from the retina demonstrated [Podoleanu 2014]. as shown whethere the majority of SB-OCT reports use linear cameras at 2070 kHz, which represents a line scan rate faster than maximum achievable by TD-OCT en face imaging using a resonant scanner at 16 kHz and which is more than 20 times faster than line scan rates in en face OCT using galvanometer scanners [Podoleanu 2014].

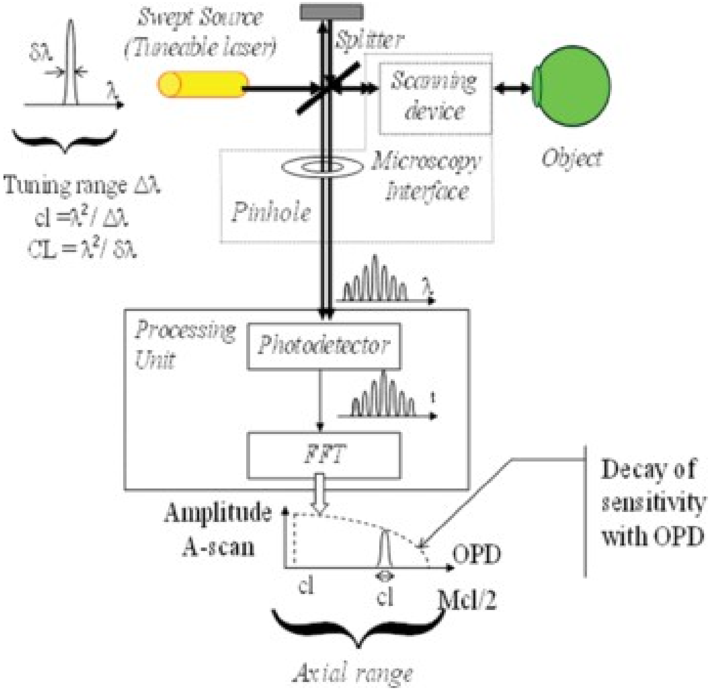


Figure 2.7: Diagram showing a typical SS-OCT setup [Podoleanu 2014]

Swept Source OCT (SS-OCT)

The Alternative to SD-OCT and SB-OCT is SS-OCT which is referred to as Swept Source OCT, this uses a photo-detector to adjust the light source to ensure the laser line, , is much more narrower than the spectral distance between adjacent peaks as shown in Figure 2.8(b).

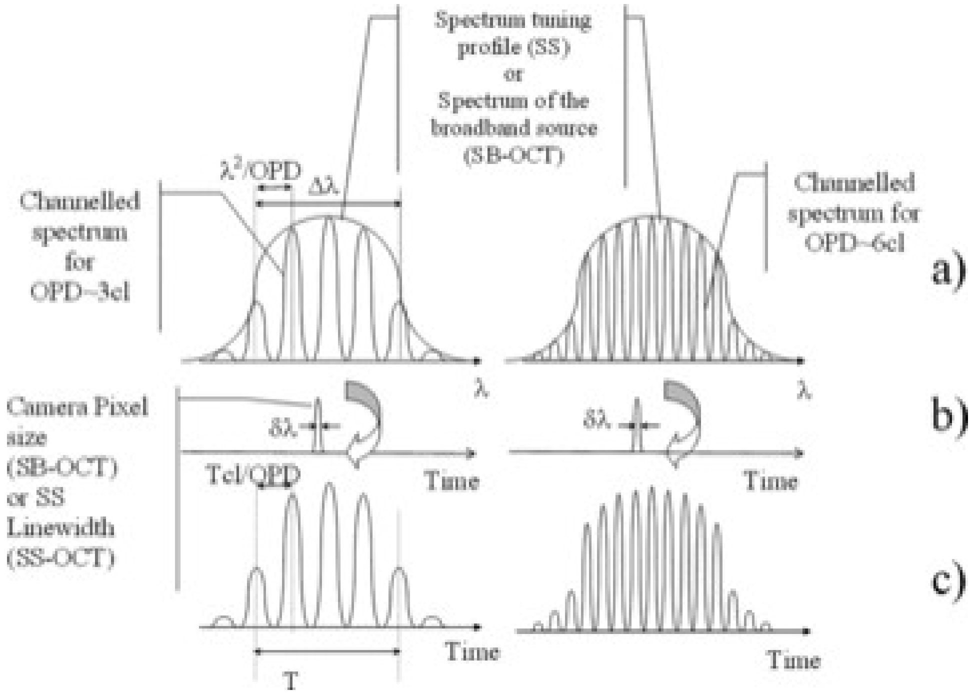

Spectrum tuning profile (SS) or Spectrum of the broadband source (SB-OCT)
λ²/OPD
Δλ
Channelled spectrum for OPD~3cl
Channelled spectrum for OPD~6cl
a)
λ
λ
Camera Pixel size (SB-OCT) or SS Linewidth (SS-OCT)
δλ
δλ
Tcl/OPD
Time
Time
b)
c)
Time
Time
T

Figure 2.8: (a) Spectrum output of the interferometer for two OPD values, left: 3 cl, right: 6 cl; (b) SD-OCT sampling function, of width , spectrum is read in a time T either by downloading the charge from the linear camera array, or by tuning the frequency of the SS, the large arrow suggests that by doing so, the spectrum shown above in (a) is transferred to the output signal shown in (c) below; (c) signal delivered by the Processing unit in Figures 2.6 and 2.7. [Podoleanu 2014]

If in the ideal case, the laser line is approximated with a Dirac delta function (infinitesimally small linewidth), then it shown that the photo-detected signal takes the exact shape of the channelled spectrum. By perform a fast Fourier transform of this signal, this translates the periodicity of the channelled spectrum into peaks of different frequency, related to the OPD. The Result is, an A-scan is obtained, as shown at the bottom of Figure 2.7,when the Object is a mirror [Podoleanu 2014]. Recent Development in the fast tunability of laser sources has revived the interest in SS-OCT. For example the time required to tune the wavelength is determined by the time to produce an A-scan [Podoleanu 2014]. As a result with tuning speeds in excess of 5 MHz making the SS-OCT the fastest scanning OCT method that has lead to sufficient quality of in vivo images acquired from tissue[Podoleanu 2014].

2.6.5 OCT Improvements: Enhanced Depth Imaging (EDI)

The choroid is an intergral part of the eye which accounts for most of the ocular blood flow,is the source of the most common intraocular tumor in adults, and yet defies current methods of clinical examination [Spaide et al. 2008]. One of the difficulties with examining the choroid, is the decrease in imaging sensitivity and resolution in relation to the increase in depth within the structure [Spaide et al. 2008]. As a result of the consequence of the wavelength used, typical OCT instruments may have results in significantly reduced resolution for retinal visualization, which ultimately may limit clinical appeal[Spaide et al. 2008. [Spaide et al. 2008] mentions a new developed approach referred as Enhanced Depth Imaging (EDI) that enhances the visualisation of the deeper layers such as the choroid, which is typically applied in conventional commercial instruments.

EDI revolves utilizes the principle in SD-OCT that by convention,objects nearer the zero delay are imaged at the top of the screen and deeper objects are imaged further down the on the screen. The result is images placed near the top of the display have greater detail than those lower in the display [Spaide et al. 2008].thus the direct result of this focused form of illumination is an increase of the imaging sensitivity of the retinal structure. On more recent and newer OCT Instruments, EDI is becoming more readily available as a capturing software option [Regatieri et al. 2012].

Unfortunately, EDI does have limitations. For instance, The wavelength used in EDI is not optimal to image the choroid,necessitating that steps be taken to improve image quality. As such OCT devices specifically designed to image the choroid undoubtedly would be able to outperform the SD OCT instruments used for this specific purpose. However, vice versa a device designed specifically for choroidal imaging would not produce optimal retinal images for any given bandwidth of the light source [Spaide et al. 2008].

Another limitation in Enhanced Depth Imaging is that to be able to obtain these images, the subject must be able to fixate with at least one eye. With unsteady fixation, it would be difficult to obtain 100 scans per section in a reasonable time. [Spaide et al. 2008].

2.6.6 Conclusion and Application

we have discussed briefly the technology known as Optical Coherence Tomography (OCT) in which details of its origins, principles and applications, and further more, the various forms of OCT. Overall, OCT is an effective method of scanning the human retina, in addition with recent developments such as EDI which has improved the visualisation of choroidal structures.

In this project, all data mentioned in this report is produced through the use of a OCT machine at Cardiff University Hospital, this particular OCT machine is a Spectral Domain-OCT (SD-OCT) variant, as this has been shown (see section 2.6.4) to be one of the best variants in detecting Choroidal diseases.

Chapter 3

Implementation

Authors Note: Since the start of the project, after the implementation of the proposed solution, the results from the experiment has shown promising results (see section 4). However, there were potential improvements that were identified which led to several possible refinements in the experiment. These refinements are discussed in section 4.2.

3.1 Proposed Solution

The Proposed solution involves taking a data of OCT images, and labelling the data, with reference to the actual classification of the data set, this know as ground truth labelling, the solution then utilises an adaptation of a standard Gabor Filter that is applied to a select sample of OCT Images within a masked region, this masked region is generated through the use of the sample of OCT images and their corresponding Labelling data. Once the filter is applied, the result is a filter response vector. A energy transform is applied to this filter response vector in order to achieve and view the energy response from the filter response vector. Finally, the features of the dataset is extracted and the process is repeated and compiled to form a data set of features for each stages of AMD. The compiled dataset is used to train the classifiers resulting in several matrices of classification results, each representing the different types of classifiers used in this experiment. This validation step is repeated for each form of validation method and is then analysed in order produce a overall average classification result (Mean average) and corresponding threshold of each result (Standard Deviation).

3.2 Methodology

This section will explain the core stages and methods used in this experiment, we have the following stages:

- Labelling - This stage primary involves labelling the provided OCT images, We decided to manually label by hand, due to the unknown nature of the Data and lack of professional criteria at the time, resulting in a lack of a feasible method to automate the labelling process. As a result, the Data was accurately labelled at the cost of resource intensive.
- Image Processing - Masking and Filtering is applied to the dataset, through the use scripts, these processes are automated, the Masking process is relatively simple and as a result, it is quite efficient, the filtering on the other hand, is rather complex and requires several stages of processing. This results the majority of the experiments run is taken up by this. item Feature Extraction - The filter response vector undergoes an energy transform stage followed by the features of the dataset being extracted, The Energy Transform Stage in particular utilizes a non linear transform in order to produce the energy response from the filtered dataset. Finally, the Featured image is extracted from the energy responses, to which the process is repeated for all patients in a single stage of Age-Related Macular Degneration.
- Training & Validation - Once all cases are processed ad their features are extracted, it is further compiled t form a dataset of features. This data is then used to train the classifier and provide test samples for validation.

3.3 Labelling

3.3.1 Labelling Principles

The choroid was primarily reviewed, and highlighted, looking for any obvious signs of AMD such as *Choriorentinal Altrophy* which makes the appearance of the choroid membrane more visible compared to the surrounding area of the choroid as finely seen in figure 3.2 or more clearly shown in figure 3.1.

Other indicators to look for is the in the disruption (or lack) of retina layers (see figure 3.1), this trait is prevalent in all forms of choroidal diseases, however there are some discerning features to differentiate between diseases. For example in the case with Neovascular Membranes (wet AMD), in the early stages there are minute differences to discern from using ophthalmoscopy and on stereographic photographs, in early stages only mild elevation is present secondary to sub retinal fluid (SRF) leakage in areas of RPE mottling these traits can be detected in OCT with possible identification of layers involved in a CNVM. CVNM can be classified depending on its appearance in relation to the RPE layer, and location in the fovea. PED can also be discerned from, as they appear beneath the RPE layer, as fibrovascular or hemorrhagic pigment epithelial detachment (see figure 3.3).

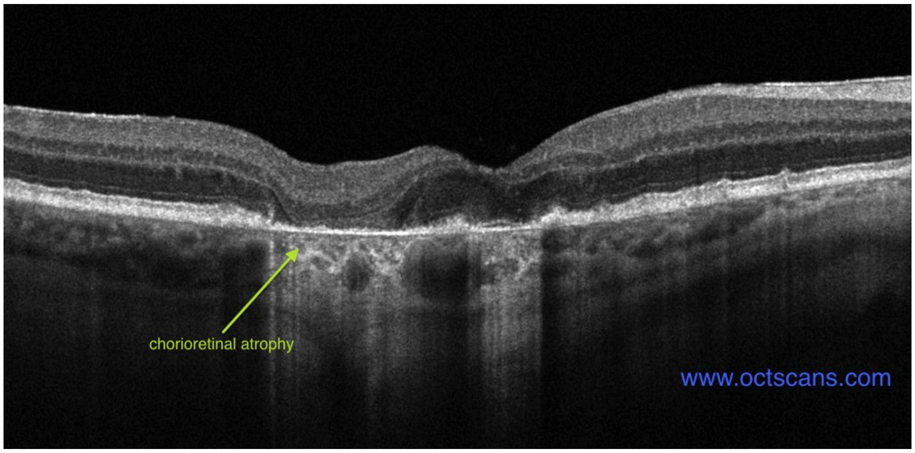


Figure 3.1: OCT Scan displaying the lost of outer retinal layers, and appearance retinal atrophy. *Source: http://www.octscans.com/age-related-macular-degeneration.html*

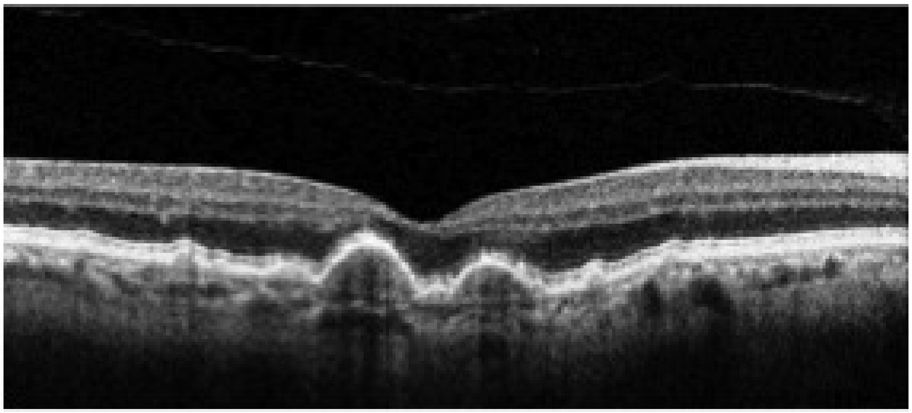


Figure 3.2: OCT Scan displaying Dry AMD with large Drusens
Source: `http://www.octscans.com/age-related-macular-degeneration.html`

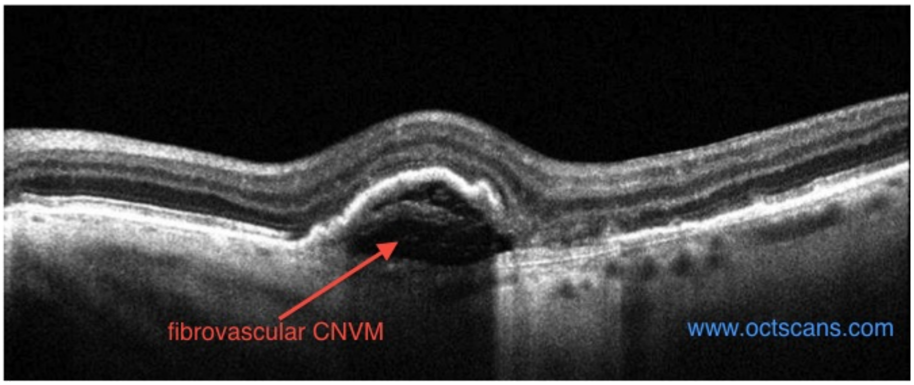


Figure 3.3: OCT Scan displaying pigment epithelial detachment secondary to a CNV
Source: `http://www.octscans.com/age-related-macular-degeneration.html`

3.3.2 Labelling

Using the mentioned principles, the OCT images are labelled with the knowledge of which set of images has a particular form of diseases (if any), this is known as the ground truth. The Process involves manually labelling the data set by taking the grand population of each disease case, and taking a sample of this population to form our test sample. This sample follows these specifications:

> **Sample Specification**
>
> Test Sample - 560 images in total
>
> 3 diseases cases and a normal case
>
> 7 patients are selected at random
>
> 100 images are labelled per patient
>
> 20 random images are selected per patient for the dataset

Using a medical viewing software,Osirix Rosset et al. 2004, the raw images were loded and viewed (at 2 images per interval) and a drawing tool feature is used to create the region of interest. By using a closed contour, it is possible to create a Region of Interest (see figure 3.4) with minimal amount of points whilst having flexibility in the shape of the contour. After the ROIS are created, ensuring the highlighted area follows the labelling principles, and that the area has minimal noise i.e. areas in the image which are completely black. The batch of labelled images are exported via a plugin named *ROI Export* to a *Comma-Separated values (csv)* file for further processing.

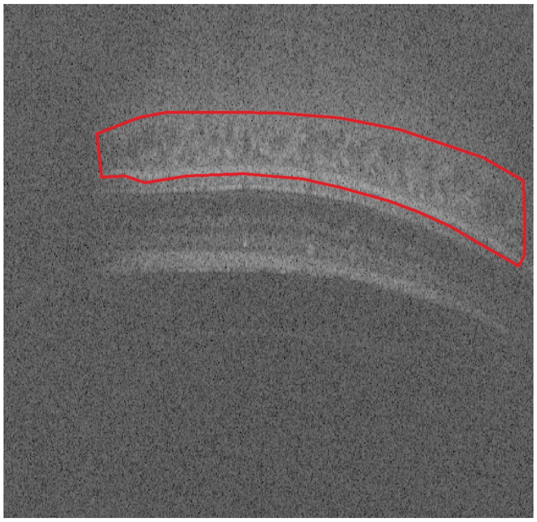


Figure 3.4: OCT scan with sample ROI highlighted

3.4 Image Processing

Once all the batches were labelled, smaller sample were created consisting of 20 random images from the aforementioned batches. This form our dataset that would be used for further stages in the experiment. The next step in the experiment can be separated in two stages, a Masking stage is applied such that all following stages specifically target the labelled region of interest, and a filtering stage where the masked images have the Gabor Filter applied to it, both are discussed in further detail below.

3.4.1 Masking

Using the gathered labelling data, a poly mask is created from the shape of the closed contour used during the labelling process. This can then applied on a image and a logical comparison is performed. This determines whether the pixel in the raw image is within the created mask or not; this can be explained in the following:

$$f(n) = \begin{cases} 0 & \text{if } n \text{ is not within the Mask} \\ 1 & \text{if } n \text{ is within the Mask} \end{cases}$$

The Result is a logical representation of a image, which can be used in later stages. In this case, it is used after the energy transform process, a logical search is used to locate positive results (i.e. 1's), the locations of the positive results are mapped to an image, now containing energy responses rather than pixel value or filter response, and the corresponding energy responses are copied to a new matrix representing the masked energy responses named, *EnergyMasked*.

3.4.2 Filtering

The Gabor filter (refer to code, provided by [Haghighat et al. 2015] in the appendix] (see Equation 3.1 in sector 3.4.3 for Gabor Filter Equation) is a linear filter frequency and orientations. This is filter forms a M x W window grid of features typically in the dimensions of *M number of scales* by *Wnumber of orientations*. The result of processing the OCT images through this filter are images which shows the frequency response for each orientation an scale being applied, in the form of a feature vector. These vectors are reshaped to form feature response images. (see figure 4.1 for a formatted output of the gabor filter). For additional clarity, to differentiate between response values per pixel in a filtered image, a heat map is also created. As the response value are in real format it was possible to have a heat map applied to the image, this in turn, applies a temperature to each pixel in correspondence to it value. The result is a image with each frequency response value as a gradient of colour going from large negative numbers as cold to large positive numbers as hot (see figure 4.2 in chapter 4).

3.4.3 Gabor Filters

$$g(t) = \mathrm{k}e^{j\theta} w(at)s(t) \tag{3.1}$$

where

$$w(t) = \mathrm{e}^{\pi t^2} \tag{3.2}$$

$$s(t) = \mathrm{e}^{j(2\pi f_o t)} \tag{3.3}$$

$$e^{j\theta} s(t)\mathrm{e}^{j(2\pi f_o t+\theta)} = (sin(2\pi f_o t + \theta), jcos(2\pi f_o t + \theta) \tag{3.4}$$

k, θ, f_o are filter parameters. We can think of the complex Gabor filters as two out of phase filters that is temporarily assigned in the real and complex part of a complex function [Movellan 2002]

Gabor filter Complex real part

$$g_r(t) = w(t)\sin(2\pi f_o t + \theta) \tag{3.5}$$

Gabor filter Complex imaginary part

$$g_i(t) = w(t)\cos(2\pi fot + \theta) \tag{3.6}$$

Frequency response

$$\hat{g}(f) = ke^{j\theta} \int_{-\infty}^{\infty} \mathrm{e}^{j2\pi f t} w(at)s(t)\mathrm{d}t \tag{3.7}$$

$$= ke^{j\theta} \int_{-\infty}^{\infty} \mathrm{e}^{j2\pi(f-f_o)t} w(at)s(t)\mathrm{d}t \tag{3.8}$$

$$= \frac{k}{a}\mathrm{e}^{j\theta}\hat{w}\left(\frac{f-f_o}{a}\right) \tag{3.9}$$

where

$$\hat{w}(f) = w(f) = \mathrm{e}^{-\pi f^2} \tag{3.10}$$

The Spatial (2-D) Gabor Filter
Formula of a complex Gabor function in space domain

$$g(x,y) = s(x,y)w_r(x,y) \tag{3.11}$$

where s(x, y) is a complex sinusoid, known as the carrier, and wr(x, y) is a 2-D Gaussian-shaped function, known as the envelope.

3.5 Feature Extraction

Once the sample of OCT images are filtered and formatted, a non linear transformation is applied, specifically the following bounded non-linearity

$$\varphi(t) = tanh(\alpha t) = -\frac{1-e^{-2\alpha t}}{1+e^{-2\alpha t'}} \tag{3.12}$$

where α is a constant. This non-linearity has some similarities to the sigmoidal activation function used in artificial neural networks. In the experiments, we have used an empirical value of $\alpha = 0.25$ which results in a rapidly saturating, threshold-like transformation [Dunn and Higgins 1995]. Through the application of the non-linearity transforms, the sinusoidal modulations in the filtered images become square modulation , which can then be used in extracting the features .

However the complex forms and signs of AMD means the energy response is clearly binary, so instead of taking each energy reponse from each patient individually, we instead compute the average absolute deviation (AAD) from the mean in small overlapping windows. This is similar to the texture energy measure that was first proposed by Laws [Laws 1980] this is given by, the feature image $e_k(z,y)$ corresponding to filtered image $r_k(x,y)$ in the following:

$$e_k(z,y) = \frac{1}{M^2} \sum_{(a,b)\in W_{xy}} |\varphi(r_k(a,b))| \tag{3.13}$$

where $\varphi(.)$ is the nonlinear function in (3.12) and W_{xy} is an M x M window centred at the pixel with coordinates (x,y). The size, M, of the averaging window in (3.13) is an important parameter. This is because reliable measurement of texture features calls for larger window sizes. in comparision, more accurate localization of region boundaries calls for smaller windows [Dunn and Higgins 1995]. In Addition, by using Gaussian weighted windows, instead of unweighted windows, the result is like to be more accurate in localization of texture boundaries (i.e. the affected area,loss of RPE etc).

For each filtered image we use a Gaussian window which has a space constant σ that is proportional to the average size of the intensity variations in the image. This is given by the following equation where in a Gabor filter the radial frequency is noted as u_0.

$$T = N_c/u_0 \text{ Pixels} \tag{3.14}$$

where N, is the width (number of columns) of the image. We found a $\sigma \approx$ 0.52T to be most reliable value in extracting the features from the energy response vector

3.6 Training and Classification

The standard Matlab 2014b machine learning toolbox and a 3rd party SVM Library [C.-C. Chang and C.-J. Lin 2011] is used to analyse the compiled data set of extracted features. This data set is composed differently based on the following cross-validation method:.

- **k-fold** - where the data set is partitioned into k equal sized sub samples. Of the k sub samples, a single sub sample is retained as the validation data for testing the model, and the remaining (k1) sub samples are used as training data. The cross-validation process is then repeated k times (the folds), with each of the k sub samples used exactly once as the validation data, the k results are then averaged (or combined) to produced a single estimation.

- **2-fold** (half-half) - this is the simplest variation of *k-fold*, where in each fold, a set of randomly assigned data points is allocated, thus both folds are of equal size (hence half-half), 1 fold is then used for training and the other for testing then followed in vice-versa, again the results are averaged and a single estimation is produced

- **Leave-p-out** (**LpO CV**) this involves using p observation for testing and the remaining observations as the training set. This process is repeated on always to cut the original sample on a validation set of p observations and a training set. In this experiment the leave-one-out (LOO) variation was only used, as a result LOO cross-validation doe not have the calculation and scaling problem of general LpO cross validation as it runs at $\mathcal{O}(n)$ complexity

Using the mentioned cross validation methods we developed the following specification:
Classifer Training and Validation Sepcification

- Given data input of features.
- Number of runs: 10 times.
- Number of permutation to the dataset : 10 (10 fold), 2(2 fold) 7(leave-one-patient-out)

each classifier is tested following this specfication, each test produces a prediction which goes through a confusion matrix, to serve two functions.Firstly to vizualize the performance of the classifers (see figure fora template of the expected confusion matrix) and secoundly to filter the result to obtain the true values, these true value are store in a result matrix which will be used to compute the overall accuracy of the classifiers

		Prediction				
		Norm	CVNM	AMD	PED	Total
Actual	Norm	a	b	c	d	$a+b+c+d$
	CVNM	b	c	d	e	$b+c+d+a$
	AMD	c	d	a	b	$c+d+a+b$
	PED	d	a	b	c	$d+a+b+c$
	Total	$a+b+c+d$	$b+c+d+a$	$c+d+a+b$	$d+a+b+c$	N

Table 3.1: A simplified example of expected Confusion Matrix

Chapter 4

Results

After processing the samples through a permutation process in order to randomize the samples that would be fed to the classifiers. it goes through a training process and a validation process. This process of creating permutations, training and validating is dependent on the validation method used at the time. After the validatio process is completed and the following predictions are produced. These results have a confusion matrix applied to it, in order to extract the true values of the results. These true values are then stored in a matrix *Nresult(ClassifernName)* to be used later. The entire testing process is repeated 10 times to ensure the test are robust and have been adequately tested, to the point where these results are not saturated leading to over fitting.

We compiled the results R from the *NResult* matriacies and compute the mean $\langle R \rangle$ and stadard dev σ. This produced the global average score and robustness for each classifier. The results in table 4.1 show promising accuracy rates, with KNN scoring **84.4%** as the lowest and SVM scoring **93.8%** as the highest except for the deep learning techniques which score significantly lower at **25%** and **20.6%** respectively this pattern continues throughout the other forms of validation method seen in table 4.2 and table4.3. These results are discussed later in the analysis section where points are made to further improve the experiment.

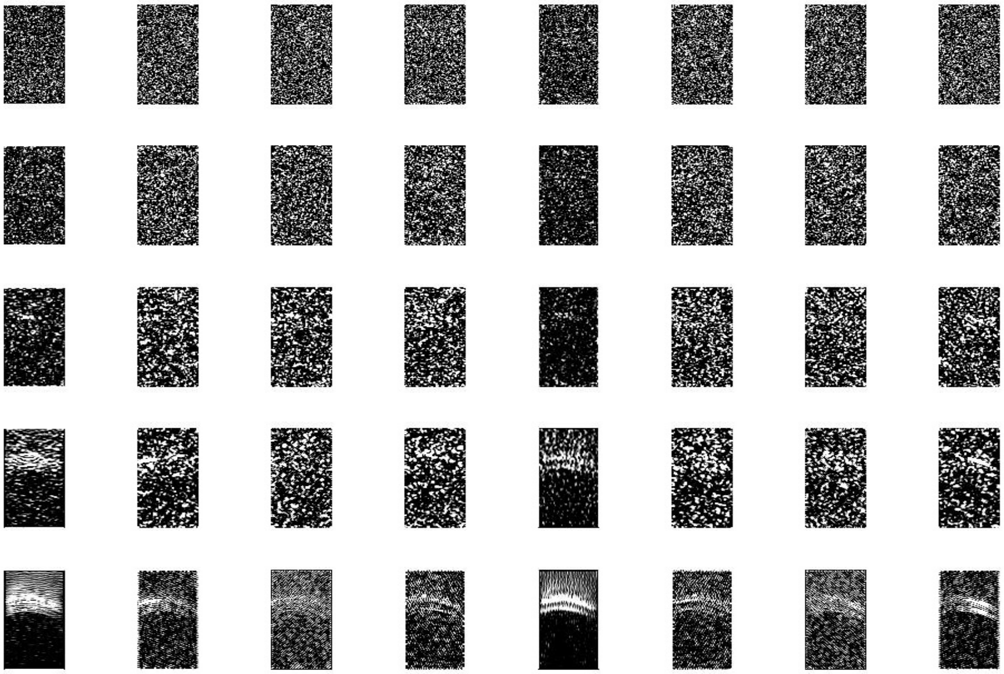


Figure 4.1: Grid of filter responses of a OCT image

Through the application of the Gabor Filter to the image sample, the output is a filtered response image. Figure 4.1 displays a window of size M X W where M is scale and W is orientations of OCT images responding to the Gabor filter. As you can see, depending the scale and orientation used, the clarity of the affected area differs it is these differences that are used to build a knowledge base for discerning the stage of AMD. We have also applied a heat map over each image of the window in figure 4.2, a heat map filter is applied to the previously filtered images, in order to show more easily, that the responses are not binary (0 — 1), but rather have real values that indicates the strength of the response to the filter in a precise format. There is indication of these characteristics have differing levels of intensity due to different gradient of colour depicted in the heat map, with colours closer to red being less pronounced and vice versa witch colours closer to yellow.

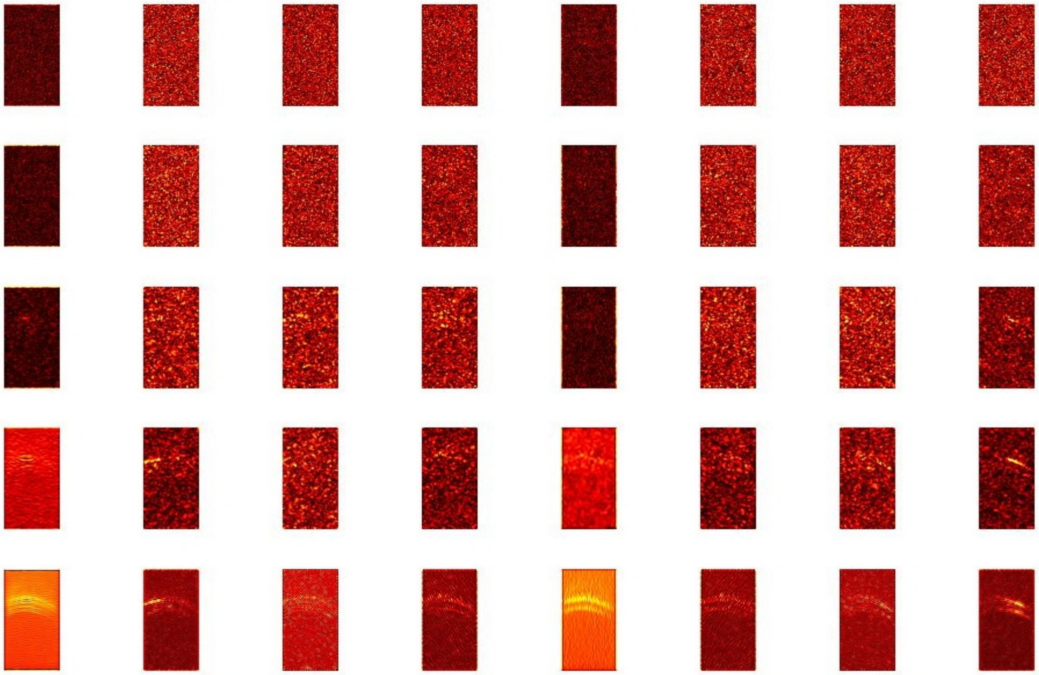


Figure 4.2: Heat map version of grid seen in figure 4.1

In both figure 4.1 and figure 4.2, there is evidence through the use a Gabor filter to the sample, that the filter responses and consequently the heat map show on a statistical perspective, there are some distinguishing characteristics, these are areas are highlighted in the proposed affected area (see figure 3.4 for reference) as areas with greater intensity or ”heat” in the mentioned figures.

4.1 Analysis

Earlier we made the claim that there are discernible characteristics between the stages of Age-Related Macular Degeneration. if we look table 4.1 and table 4.2 below, there is evidence to support thaat claim made in regards between the filter response image in figure 4.1 and the heat map version,figure 4.2, As the high accuracy rates between these table indicates that there are properties in these signs of AMD that can be statistically differentiated and thus classified with relatively high success.

However, if we take note of the results during a **Leave-One-Patient-Out** case or the simulated case of introducing a new patient seen in table 4.3, the results are significantly lower, this could be explained by several reasons, for example, the human body is never exact to each other. As a result, it is possible that there is a significant variance in the appearance of the affected choroidal area and consequently the indication of different stages of Age-Related Macular Degeneration would also vary between patients. If this is the case, this raises some questions on how to improve the accuracy for new patients and/or determine the key factors that affect accuracy of the classifiers. Also, there are questions regarding the results from the Restricted Boltzmann Machine and Deep Belief Network namely; why both have scored significantly lower than their traditional counterpart and are their any patterns that can be seen in order to improve the accuracy of these deep learning techniques. All of these and other possible technical improvement are mentioned in the following sections

4.1.1 10 Fold

KNN 84.4%

	Norm	CVNM	AMD	PED
Norm	0.895(±.082)	0.012	0.050	0.051
CVNM	0.038	0.818(±.086)	0.065	0.106
AMD	0.001	0.008	0.822(±.110)	0.003
PED	0.064	0.161	0.065	0.841(±.085)

RDF 87.5%

	Norm	CVNM	AMD	PED
Norm	0.920(±.080)	0.025	0.068	0.058
CVNM	0.002	0.913(±.077)	0.061	0.072
AMD	0.014	0.005	0.804(±.132)	0.006
PED	0.066	0.060	0.068	0.865(±.100)

SVM 93.8%

	Norm	CVNM	AMD	PED
Norm	0.969(±.049)	0.006	0.001	0.019
CVNM	0.004	0.930(±.062)	0.029	0.066
AMD	0.002	0.007	0.940(±.074)	0.001
PED	0.027	0.059	0.032	0.916(±.064)

RBM 25.0%

	Norm	CVNM	AMD	PED
Norm	0.170(±.376)	0.170	0.170	0.170
CVNM	0.320	0.320(±.467)	0.320	0.320
AMD	0.200	0.200	0.200(±.400)	0.200
PED	0.310	0.310	0.310	0.310(±.463)

DBN 20.6%

	Norm	CVNM	AMD	PED
Norm	0.238(±.424)	0.240	0.236	0.239
CVNM	0.168	0.167(±.370)	0.168	0.170
AMD	0.000	0.000	0.000(±.000)	0.000
PEDM	0.420	0.421	0.420	0.421(±.375)

Table 4.1: 10 Fold Results

4.1.2 2 Fold

	KNN 82.4%			
	Norm	CVNM	AMD	PED
Norm	0.899(±.052)	0.029	0.072	0.07
CVNM	0.044	0.815(±.057)	0.085	0.122
AMD	0.006	0.02	0.787(±.041)	0.011
PED	0.053	0.138	0.058	0.798(±.054)

	RDF 77.8%			
	Norm	CVNM	AMD	PED
Norm	0.858(±.065)	0.069	0.1	0.142
CVNM	0.012	0.823(±.063)	0.081	0.114
AMD	0.036	0.02	0.705(±.057)	0.018
PED	0.096	0.09	0.115	0.728(±.056)

	SVM 90.0%			
	Norm	CVNM	AMD	PED
Norm	0.936(±.031)	0.017	0.015	0.054
CVNM	0.010	0.903(±.037)	0.058	0.069
AMD	0.005	0.015	0.888(±.034)	0.003
PED	0.050	0.068	0.041	0.876(±.051)

	RBM 25.3%			
	Norm	CVNM	AMD	PED
Norm	0.350(±.477)	0.348	0.338	0.350
CVNM	0.200	0.200(±.400)	0.200	0.200
AMD	0.051	0.053	0.063(±.223)	0.05
PED	0.400	0.400	0.400	0.400(±.490)

	DBN 24.7%			
	Norm	CVNM	AMD	PED
Norm	0.215(±.398)	0.226	0.218	0.226
CVNM	0.200	0.200(±.400)	0.200	0.200
AMD	0.150	0.150	0.149(±.354)	0.150
PED	0.436	0.425	0.434	0.425(±.482)

Table 4.2: 2 Fold Results

4.1.3 Leave-One-Patient-Out (LOVop)

KNN 29.7%				
	Norm	CVNM	AMD	PED
Norm	0.449(±.371)	0.159	0.162	0.361
CVNM	0.299	0.402(±.283)	0.295	0.405
AMD	0.015	0.073	0.189(±.363)	0.087
PED	0.238	0.368	0.355	0.148(±.178)

RDF 41.7%				
	Norm	CVNM	AMD	PED
Norm	0.512(±.324)	0.117	0.205	0.377
CVNM	0.093	0.590(±.379)	0.175	0.305
AMD	0.127	0.037	0.298(±.344)	0.051
PED	0.27	0.258	0.323	0.268(±.261)

SVM 39.6%				
	Norm	CVNM	AMD	PED
Norm	0.487(±.267)	0.134	0.235	0.413
CVNM	0.069	0.521(±.326)	0.222	0.316
AMD	0.148	0.109	0.360(±.462)	0.055
PED	0.298	0.238	0.185	0.218(±.232)

RBM 25.0%				
	Norm	CVNM	AMD	PED
Norm	0.329(±.470)	0.329	0.329	0.329
CVNM	0.129	0.129(±.335)	0.129	0.129
AMD	0.286	0.286	0.286(±.452)	0.286
PED	0.258	0.258	0.258	0.258(±.458)

DBN 25.4%				
	Norm	CVNM	AMD	PED
Norm	0.186(±.389)	0.186	0.186	0.186
CVNM	0.299	0.300(±.459)	0.286	0.294
AMD	0.258	0.258	0.272(±.445)	0.264
PED	0.259	0.258	0.258	0.258(±.438)

Table 4.3: Leave-one-Patient-out Results

By analysing the results and resultant images that have been shown alone, there endless possible combinations of factors to consider. We have determined that the variance between patient is a core problem, that relies on a improved textual analysis solution. To resolve this problem we have devised several refinements that can be both to the experiment on a technical standpoint and on several principles of the experiment. These two categories of refinement are the by-product of conducting this experiment as we have found by using one approach such as limiting the sample size to 7 patients per case, we have not explored the possibility of using more patients instead. Another example is the application of one type of feature via the Gabor filter, instead it may be possible to achieve better results with different features or a combination of features. as you can see from these example there are numerous possibilities, which can be used for future work. The following section discusses the on other possible refinements that we have discovered, with evidence showing some have started to be implemented.

4.2 Possible Refinements

As previously mentioned, one of the accountable causes for the drop in accuracy when a new case is introduced is the variance in the appearance of the different stages of Age-Related Macular Degeneration. In such cases there are numerous possible combinations to resolve this problem. This section will discuss a few examples that aims to provide and understanding of the problem and hints of potential areas for future development.

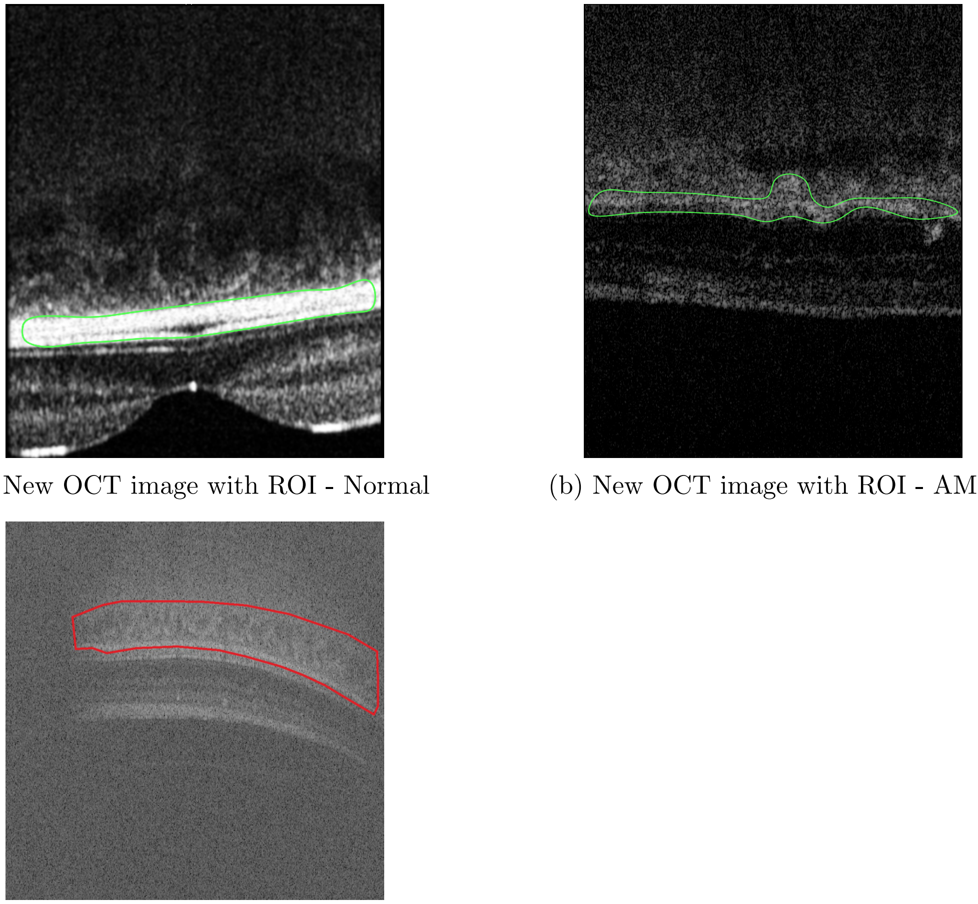
4.2.1 Higher Quality Images

Due to a time constraint and insufficient resources, it was not possible to perform a full experiment on a recently developed data set, that provided significantly higher resolution OCT images. This would have potentially allowed the identification of more minute details, As evidence there is a reduction in the amount of noise that accompanied with the older OCT images. In addition, a clearer guide was provided by a specialist to aid in labelling the key signs of choroidal diseases. This resulted in regions of interest being more focused and having more contrast in comparison to the old data set. This can be seen in this comparison between (figure 4.3a) and (figure 4.3b) to (figure 4.3c) below.

4.2.2 Adjustment of image window size

Another possible improvement would be the adjusting the window (brightness & contrast) size of the Image Sample. This is particular meaningful as only a range of pixel intensities are viewed, resulting in the image having a clearer distinction between strong pixel intensities and weaker ones. A stronger distinction between pixel intensities could cause the filter response and consequently the energy response to affected differently to areas where signs of AMD are more pronounced in the image. Consequently, the training and accuracy of the classifier would be affected, possibly resulting in significant changes to the original result. An example of this is seen in (figure 4.3a) and (figure 4.3b) which have significantly higher contrast compared to (figure 4.3c) below. This is because the new images are viewed with an appropriate window level whilst the old image is seen in its raw window size.

Figure 4.3: Comparision between new and old OCT images



(a) New OCT image with ROI - Normal

(b) New OCT image with ROI - AMD

(c) old OCT with ROI

4.2.3 Utilising different Scanning Methods

it has been mentioned in section 2.6, that there are numerous forms of OCT Scanning, it is possible with the development of higher specification OCT scanner. Different combinations of parameters can be used to improve on the scanning quality. This was briefly hinted in section 4.2.1 with the introduction to higher quality images. These images were produced through the combination of higher quality OCT scanning techniques and the use of a new light source. In relation to the scanning methods, future development in both technology and practice may also counter the problem that occurs when scanning a patient namely, the micro movements and agitations that occur in the scanning process. As a result of these agitations, the image quality is some time indiscernible or partially visible

4.2.4 New Features

In this experiment only the energy response from a Gabor filter is used for feature extraction, this is because a review of different features fall out of the scope of this project, and a time constraint was imposed, making a full review difficult. This means the use of different features or possibly a combination of features may lead to more interesting results. In our experiment, the use and analysis of a Gabor filter has shown there is evidence that support the stages of Age-Related Macular Degeneration are statistically discernible to a degree. In regard to the variance between patients, in order to look for the symptom or ”signs” of the stages of AMD, texture analysis plays a key role, in developing an algorithm that discerns visual patterns between the stages.

A particularly interesting factor to consider is the use of Deep Learning algorithm. Despite the RBM and DBM scoring rather poorly in comparison to their traditional machine learning counterparts. It should be noted by looking at the overall classification between each validation method, both RBM and DBN made improvements in their accuracy, This is further supported with the introduction of a patient, where both classifier either retained or improved in accuracy. These piece of evidence support the theory that Deep Learning may be further used to improve the classification of Age-Related Macular Degeneration. A generalised theory on why RBM and DBN score so low, may be the fact that deep learning algorithm require significantly larger knowledge databases to fully ”learn” the properties of each stages of AMD. This is further discussed in later is section 4.2.6

4.2.5 Labelling Criterion

As this experiment relies heavily on the manual classification and labelling of the dataset. The implementation of a more refined labelling criterion would great benefit the labelling process due to creating regions of interests that are more focussed on the particular areas where the signs of Age-Related Macular Degeneration have been shown. This can be achieved by seeking professional consult with a medical specialist in the field, in order to create a more focussed criterion, such as criteria may include; the specific areas of interest, particular signs of each stage of AMD, possible sizes of infected region etc. This has started to be implemented in this experiment, with the more focussed labelling seen in (figure 4.3a) and (figure 4.3b).

4.2.6 More Patients

A simple observation of our experiment specification can find that our sample sizes are quite limited, it is possible to use a larger set of data samples, a suggestion would be to utilize the full amount of data per patient. In this case since in each patient, there is 512 images, it is possible to label the entire set of image at small intervals, this way only a fraction of the entire dataset is used, but is still a fully comprehensive representation of that particular patient. This result in more data having to processed both manually and computationally however the results may benefit the experiment overall, as a more comprehensive dataset would produce a more detailed collection of feature. Another expansion of this is adding more patients, again looking at our experiment, only 7 patients were used in each case, by adding more patients it is possible to create a larger collection of features, which can be used to enhance the training of the classifiers. This is because machine learning requires a knowledge base in order to determine a classification. By using larger knowledge base, the generalised theory that more features from a new patient would be discernible would seem plausible. This is especially true for deep learning network which require large knowledge base in order build a core network, and then expanding the network to include feature which are have degrees of similarity to the features in the core network.

Chapter 5

Conclusion

We have presented an approach using a combination of texture analysis, retinal image analysis and machine learning to explore the possibility of creating an automated classifier for the stages of Age-Related Macular Degeneration. Our proposed approach uses a developed criterion on how the OCT images are treated before feature extraction, several algorithms were developed or used to aid in the feature extraction of the OCT images, and in the training of the classifier. The texture features extraction algorithm utilises an adaptation of a standard Gabor Filter. This choice in using the Gabor Filter and it parameters was influence by several sources [T. Chang and Kuo 1993; Jain and Farrokhnia 1990], that utilize Gabor filters in similar texture analysis problems. The materials for the Gabor filter and the use of a non-linearity algorithm were suggested by [Dunn and Higgins 1995; Movellan 2002] and [Jain and Farrokhnia 1990] respectively. We showed that the use of a Gabor Filter and non-linear transformation was sufficient to be used for feature extraction. We have also presented an algorithm that forms a test suite incorporating several traditional machine techniques including a couple of the more recently developed deep learning techniques. This paired with our use of the 10 fold, 2 fold and leave-one-patient out validation set and analysis algorithm has provided classification result that has shown that automated classification of choroidal diseases is a valid option for most cases.

The experiment fell short with regard to the robustness to adapt to new patients due to the variance between patients. This may the result of the limitation of the experiment, as many possibilities such as variable sample size, different perspective to labelling the OCT image etc that were not explored fully during this experiment, as such there is much that could be improved from this basic experiment. Another notable limitation is the lack of criterion and exploration in the parameters of several algorithm name the Gabor Filter scale and orientation As well as the criterion in choosing the value of σ in the non-linear transform algorithm. These values were taken from past experiments or theories that were similar to our experiment, thus an assumption was made that the standard parameters used would be sufficient.

Our experiment took several liberties in the experiment such as limiting the sample size, and the use of only type of feature. The experiment also take the assumption that the data collected has already been classified, however this may not always be the case as such the development of the knowledge base is still resource intensive. In the end, we achieved our goal for this project. However, we believe by utilising several key improvements that are not in the scope of this experiment such as including a larger data sample, using other types of features and having a better criterion in the labelling process. These improvements and several other mentioned in this report could aid in remedying the underlining issue of there being variance between patients. This in turn could potentially lead to the development of a robust automated classifier, making it a viable option in research and medicinal diagnosis use.

Bibliography

A. Monadjemi M. Mirmehdi, B. T. "Restructured eigenfilter matching for novelty detection in random textures". In: (cit. on p. 16).

Ali, M. and R. Parlapalli (2010). "Signal processing overview of optical coherence tomography systems for medical imaging". In: Texas Instruments, SPRABB9–June (cit. on p. 31).

Altman, N. S. (1992). "An introduction to kernel and nearest-neighbor nonparametric regression". In: The American Statistician 46.3, pp. 175–185 (cit. on p. 17).

Altmann, A., L. Toloşi, O. Sander, and T. Lengauer (2010). "Permutation importance: a corrected feature importance measure". In: Bioinformatics 26.10, pp. 1340–1347 (cit. on p. 19).

Amit, Y. and D. Geman (1997). "Shape quantization and recognition with randomized trees". In: Neural computation 9.7, pp. 1545–1588 (cit. on p. 18).

Bengio, Y. (2009). "Learning deep architectures for AI". In: Foundations and trends® in Machine Learning 2.1, pp. 1–127 (cit. on p. 22).

Bengio, Y., A. Courville, and P. Vincent (2013). "Representation learning: A review and new perspectives". In: Pattern Analysis and Machine Intelligence, IEEE Transactions on 35.8, pp. 1798–1828 (cit. on p. 22).

Bengio, Y., P. Lamblin, D. Popovici, H. Larochelle, et al. (2007). "Greedy layer-wise training of deep networks". In: Advances in neural information processing systems 19, p. 153 (cit. on p. 24).

Bird, A. C., N. M. Bressler, S. B. Bressler, I. H. Chisholm, G. Coscas, M. D. Davis, P. T. de Jong, C. C. Klaver, B. E. Klein, and R. Klein (1995). "An international classification and grading system for age-related maculopathy and age-related macular degeneration. The International ARM Epidemiological Study Group." In: Survey of ophthalmology 39.5, pp. 367–74 (cit. on p. 26).

Bishop, C. (2006). Pattern recognition and machine learning (cit. on p. 16).

Boser, B., I. Guyon, and V. Vapnik (1992). "A training algorithm for optimal margin classifiers". In: ...of the fifth annual workshop on ... (Cit. on p. 21).

Breiman, L. (2001). "Random forests". In: Machine learning 45.1, pp. 5–32 (cit. on pp. 18, 19).

Bressler, N. and S. Bressler (1989). "The grading and prevalence of macular degeneration in Chesapeake Bay watermen". In: Archives of ... (Cit. on p. 26).

Carreira-Perpinan, M. A. and G. E. Hinton (2005). "On contrastive divergence learning". In: Proceedings of the tenth international workshop on artificial intelligence and statistics. Citeseer, pp. 33–40 (cit. on pp. 23, 24).

Chang, C.-C. and C.-J. Lin (2011). "LIBSVM: A library for support vector machines". In: ACM Transactions on Intelligent Systems and Technology 2 (3). Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`, 27:1–27:27 (cit. on pp. 12, 46).

Chang, T. and C. J. Kuo (1993). "Texture analysis and classification with tree-structured wavelet transform". In: Image Processing, IEEE Transactions on 2.4, pp. 429–441 (cit. on p. 59).

Christen, W. G., R. J. Glynn, J. E. Manson, U. A. Ajani, and J. E. Buring (1996). "A prospective study of cigarette smoking and risk of age-related macular degeneration in men". In: Jama 276.14, pp. 1147–1151 (cit. on p. 27).

Coates, A., A. Y. Ng, and H. Lee (2011). "An analysis of single-layer networks in unsupervised feature learning". In: International conference on artificial intelligence and statistics, pp. 215–223 (cit. on p. 23).

Coleman, H. R., C.-C. Chan, F. L. Ferris, and E. Y. Chew (2008). "Age-related macular degeneration." In: Lancet (London, England) 372.9652, pp. 1835–45 (cit. on pp. 26, 27).

Cortes, C. and V. Vapnik (1995). "Support-vector networks". In: Machine Learning 20.3, pp. 273–297 (cit. on p. 21).

Cover, T. M. and P. E. Hart (1967). "Nearest neighbor pattern classification". In: Information Theory, IEEE Transactions on 13.1, pp. 21–27 (cit. on p. 17).

Cruickshanks, K. (1997). "The prevalence of age-related maculopathy by geographic region and ethnicity: the Colorado-Wisconsin Study of Age-Related Maculopathy". In: Archives of ... (Cit. on p. 26).

Delcourt, C. (1998). "Smoking and age-related macular degeneration: the POLA Study". In: Archives of ... (Cit. on pp. 26, 27).

Deng, H., G. Runger, and E. Tuv (2011). "Bias of importance measures for multi-valued attributes and solutions". In: Artificial Neural Networks and Machine Learning–ICANN 2011, pp. 293–300 (cit. on p. 19).

Deng, L. and D. Yu (2014). "Deep learning: methods and applications". In: Foundations and Trends in Signal Processing 7.3–4, pp. 197–387 (cit. on p. 22).

Dietterich, T. G. (2000). "An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization". In: Machine learning 40.2, pp. 139–157 (cit. on p. 18).

Drexler, W. (2004). “Ultrahigh-resolution optical coherence tomography”. In: Journal of biomedical optics (cit. on pp. 32–34).

Drexler, W., U. Morgner, F. Kärtner, C. Pitris, S. Boppart, X. Li, E. Ippen, and J. Fujimoto (1999). “In vivo ultrahigh-resolution optical coherence tomography”. In: Optics letters 24.17, pp. 1221–1223 (cit. on p. 34).

Drexler, W. and J. G. Fujimoto (2008). “State-of-the-art retinal optical coherence tomography.” In: Progress in retinal and eye research 27.1, pp. 45–88 (cit. on pp. 32, 34).

Drexler, W., H. Sattmann, B. Hermann, T. H. Ko, M. Stur, A. Unterhuber, C. Scholda, O. Findl, M. Wirtitsch, J. G. Fujimoto, and A. F. Fercher (2003). “Enhanced visualization of macular pathology with the use of ultrahigh-resolution optical coherence tomography.” In: Archives of ophthalmology (Chicago, Ill. : 1960) 121.5, pp. 695–706 (cit. on p. 32).

Dunn, D. and W. E. Higgins (1995). “Optimal Gabor filters for texture segmentation”. In: Image Processing, IEEE Transactions on 4.7, pp. 947–964 (cit. on pp. 45, 46, 59).

Ergun, E., B. Hermann, M. Wirtitsch, A. Unterhuber, T. H. Ko, H. Sattmann, C. Scholda, J. G. Fujimoto, M. Stur, and W. Drexler (2005). “Assessment of central visual function in Stargardt’s disease/fundus flavimaculatus with ultrahigh-resolution optical coherence tomography.” In: Investigative ophthalmology & visual science 46.1, pp. 310–6 (cit. on p. 32).

Fercher, A. F., C. K. Hitzenberger, W. Drexler, G. Kamp, and H. Sattmann (1993). “In vivo optical coherence tomography.” In: American journal of ophthalmology 116.1, pp. 113–4 (cit. on p. 28).

Fercher, A. F., K. Mengedoht, and W. Werner (1988). “Eye-length measurement by interferometry with partially coherent light”. EN. In: Optics Letters 13.3, p. 186 (cit. on p. 28).

Fercher, A. F. (2010). “Optical coherence tomography–development, principles, applications”. In: Zeitschrift für Medizinische Physik 20.4, pp. 251–276 (cit. on p. 31).

Friedman, D., J. Katz, and N. Bressler (1999). “Racial differences in the prevalence of age-related macular degeneration: The Baltimore eye survey11The authors have no proprietary interest in any of the”. In: Ophthalmology (cit. on p. 26).

Friedman, J., T. Hastie, and R. Tibshirani (2001). The elements of statistical learning. Vol. 1. Springer series in statistics Springer, Berlin (cit. on p. 17).

Fujimoto, J. G. (2001). “Optical coherence tomography”. In: Comptes Rendus de l’Académie des Sciences-Series IV-Physics 2.8, pp. 1099–1111 (cit. on pp. 29–33).

Fujimoto, J. G., C. Puliafito, R. Margolis, A. Oseroff, S. De Silvestri, and E. Ippen (1986). “Femtosecond optical ranging in biological systems”. In: Optics letters 11.3, pp. 150–152 (cit. on p. 28).

Gabriele, M. L., G. Wollstein, H. Ishikawa, L. Kagemann, J. Xu, L. S. Folio, and J. S. Schuman (2011). “Optical coherence tomography: history, current status, and laboratory work". In: Investigative ophthalmology & visual science 52.5, pp. 2425–2436 (cit. on p. 34).

Glauner, P. O. (2015). "Deep Convolutional Neural Networks for Smile Recognition". In: arXiv preprint arXiv:1508.06535 (cit. on p. 22).

Gottfredsdottir, M. S., T. Sverrisson, D. C. Musch, and E. Stefánsson (1999). "Age related macular degeneration in monozygotic twins and their spouses in Iceland". In: Acta Ophthalmologica Scandinavica 77.4, pp. 422–425 (cit. on p. 27).

Group, A.-R. E. D. S. R. (2000). "Risk factors associated with age-related macular degeneration: a case-control study in the age-related eye disease study: Age-Related Eye Disease Study Report". In: Ophthalmology (cit. on p. 26).

Group, E. D. C.-C. S. (1992). "Risk factors for neovascular age-related macular degeneration". In: Archives of ... (Cit. on p. 26).

Haddad, S., C. A. Chen, S. L. Santangelo, and J. M. Seddon (2006). "The genetics of age-related macular degeneration: a review of progress to date". In: Survey of ophthalmology 51.4, pp. 316–363 (cit. on p. 27).

Haghighat, M., S. Zonouz, and M. Abdel-Mottaleb (2015). "CloudID: Trustworthy cloud-based and cross-enterprise biometric identification". In: Expert Systems with Applications 42.21, pp. 7905–7916 (cit. on p. 44).

Heiba, I. M., R. C. Elston, B. E. Klein, and R. Klein (1994). "Sibling correlations and segregation analysis of age-related maculopathy: The beaver dam eye study". In: Genetic epidemiology 11.1, pp. 51–67 (cit. on p. 27).

Hinton, G. (2010). "A practical guide to training restricted Boltzmann machines". In: Momentum 9.1, p. 926 (cit. on p. 23).

Hinton, G. E. (2009). "Deep belief networks". In: Scholarpedia 4.5, p. 5947 (cit. on p. 24).

Hinton, G. E. and R. R. Salakhutdinov (2006). "Reducing the dimensionality of data with neural networks". In: Science 313.5786, pp. 504–507 (cit. on pp. 23–25).

— (2009). "Replicated softmax: an undirected topic model". In: Advances in neural information processing systems, pp. 1607–1614 (cit. on p. 23).

Ho, T. K. (1995). "Random decision forests". In: Document Analysis and Recognition, 1995., Proceedings of the 3rd Intl Conf on. Vol. 1. IEEE, pp. 278–282 (cit. on p. 18).

— (1998). "The random subspace method for constructing decision forests". In: Pattern Analysis and Machine Intelligence, IEEE Transactions on 20.8, pp. 832–844 (cit. on p. 18).

Hsu, C.-W., C.-C. Chang, C.-J. Lin, et al. (2003). A practical guide to support vector classification (cit. on p. 22).

Huang, D., E. Swanson, C. Lin, J. Schuman, W. Stinson, W. Chang, M. Hee, T. Flotte, K. Gregory, C. Puliafito, and al. Et (1991). "Optical coherence tomography". In: Science 254.5035, pp. 1178–1181 (cit. on pp. 28, 31).

Hyman, L. and A. Lilienfeld (1983). "Senile macular degeneration: a case-control study". In: American journal of ... (Cit. on p. 26).

Jain, A. K. and F. Farrokhnia (1990). “Unsupervised texture segmentation using Gabor filters”. In: Systems, Man and Cybernetics, 1990. Conference Proceedings., IEEE International Conference on. IEEE, pp. 14–19 (cit. on p. 59).

Khan, J., D. Thurlby, H. Shahid, D. Clayton, J. Yates, M. Bradley, A. Moore, and A. Bird (2006). “Smoking and age related macular degeneration: the number of pack years of cigarette smoking is a major determinant of risk for both geographic atrophy and choroidal neovascularisation”. In: British Journal of Ophthalmology 90.1, pp. 75–80 (cit. on p. 27).

Kim, S.-W., J. Oh, S.-S. Kwon, J. Yoo, and K. Huh (2011). “Comparison of choroidal thickness among patients with healthy eyes, early age-related maculopathy, neovascular age-related macular degeneration, central serous chorioretinopathy, and polypoidal choroidal vasculopathy.” In: Retina (Philadelphia, Pa.) 31.9, pp. 1904–11 (cit. on p. 26).

Klaver, C. C., R. C. Wolfs, J. J. Assink, C. M. van Duijn, A. Hofman, and P. T. de Jong (1998). “Genetic risk of age-related maculopathy: population-based familial aggregation study”. In: Archives of ophthalmology 116.12, pp. 1646–1651 (cit. on p. 27).

Klein, M. L., W. M. Mauldin, and V. D. Stoumbos (1994). “Heredity and age-related macular degeneration: observations in monozygotic twins”. In: Archives of ophthalmology 112.7, pp. 932–937 (cit. on p. 27).

Klein, R., B. E. Klein, K. L. Linton, and D. L. DeMets (1993). “The Beaver Dam Eye Study: the relation of age-related maculopathy to smoking.” In: American journal of epidemiology 137.2, pp. 190–200 (cit. on p. 26).

Klein, R., B. Klein, and K. Linton (1992). “Prevalence of age-related maculopathy: the Beaver Dam Eye Study”. In: Ophthalmology (cit. on p. 26).

Klein, R., B. E. Klein, M. D. Knudtson, T. Y. Wong, M. F. Cotch, K. Liu, G. Burke, M. F. Saad, and D. R. Jacobs (2006). “Prevalence of age-related macular degeneration in 4 racial/ethnic groups in the multi-ethnic study of atherosclerosis”. In: Ophthalmology 113.3, pp. 373–380 (cit. on p. 27).

Ko, T. H., J. G. Fujimoto, J. S. Schuman, L. A. Paunescu, A. M. Kowalevicz, I. Hartl, W. Drexler, G. Wollstein, H. Ishikawa, and J. S. Duker (2005). “Comparison of ultrahigh- and standard-resolution optical coherence tomography for imaging macular pathology.” In: Ophthalmology 112.11, 1922.e1–15 (cit. on p. 32).

Koizumi, H., T. Yamagishi, T. Yamazaki, R. Kawasaki, and S. Kinoshita (2011). “Subfoveal choroidal thickness in typical age-related macular degeneration and polypoidal choroidal vasculopathy.” In: Graefe’s archive for clinical and experimental ophthalmology 249.8, pp. 1123–8 (cit. on pp. 25, 26).

Kokkinakis, J. (2012). Diseases and Disorders of the Choroid — Eyes and Vision Specialist Australia (cit. on p. 8).

Kozak, A., B. H. Feldman, F. M.D.Dale, R. Ed.D., M.B.A., A. Team, V. A. Shah, and WikiWorks (2015). Optical Coherence Tomography - EyeWiki (cit. on pp. 16, 28, 32).

Kumar, A. (2003). "Neural network based detection of local textile defects". In: Pattern Recognition 36.7, pp. 1645–1659 (cit. on p. 16).

Larochelle, H. and Y. Bengio (2008). "Classification using discriminative restricted Boltzmann machines". In: Proceedings of the 25th international conference on Machine learning. ACM, pp. 536–543 (cit. on p. 23).

Laws, K. I. (1980). Textured image segmentation. Tech. rep. DTIC Document (cit. on p. 45).

Leeuwen, R. van, S. Boekhoorn, J. R. Vingerling, J. C. Witteman, C. C. Klaver, A. Hofman, and P. T. de Jong (2005). "Dietary intake of antioxidants and risk of age-related macular degeneration". In: Jama 294.24, pp. 3101–3107 (cit. on p. 27).

Liaw, A., M. Wiener, L. Breiman, and A. Cutler (2009). "Package randomForest". In: Retrieved December 12 (cit. on pp. 18, 19).

Lin, Y. and Y. Jeon (2006). "Random forests and adaptive nearest neighbors". In: Journal of the American Statistical Association 101.474, pp. 578–590 (cit. on pp. 19, 20).

McCarty, C. and B. Mukesh (2001). "Risk factors for age-related maculopathy: the Visual Impairment Project". In: Archives of ... (Cit. on p. 26).

Meyers, S. M., T. Greene, and F. A. Gutman (1995). "A twin study of age-related macular degeneration". In: American journal of ophthalmology 120.6, pp. 757–766 (cit. on p. 27).

Mills, P. (2011). "Efficient statistical classification of satellite measurements". In: International Journal of Remote Sensing 32.21, pp. 6109–6132 (cit. on p. 17).

Mitchell, P., W. Smith, K. Attebo, and J. Wang (1995). "Prevalence of age-related maculopathy in Australia: the Blue Mountains Eye Study". In: Ophthalmology (cit. on p. 26).

Movellan, J. R. (2002). "Tutorial on Gabor filters". In: Open Source Document (cit. on pp. 44, 59).

Patton, N., T. M. Aslam, T. MacGillivray, I. J. Deary, B. Dhillon, R. H. Eikelboom, K. Yogesan, and I. J. Constable (2006). "Retinal image analysis: concepts, applications and potential." In: Progress in retinal and eye research 25.1, pp. 99–127 (cit. on p. 7).

Podoleanu, A. G. (2014). "Optical coherence tomography". In: The British journal of radiology (cit. on pp. 35–37).

Považay, B., B. Hermann, A. Unterhuber, B. Hofer, H. Sattmann, F. Zeiler, J. E. Morgan, C. Falkner-Radler, C. Glittenberg, S. Blinder, et al. (2007). "Three-dimensional optical coherence tomography at 1050nm versus 800nm in retinal pathologies: enhanced performance and choroidal penetration in cataract patients". In: Journal of biomedical optics 12.4, pp. 041211–041211 (cit. on p. 34).

Regatieri, C. V., L. Branchini, J. G. Fujimoto, and J. S. Duker (2012). “Choroidal imaging using spectral-domain optical coherence tomography.” In: Retina (Philadelphia, Pa.) 32.5, pp. 865–76 (cit. on pp. 25, 38).

Rosset, A., L. Spadola, and O. Ratib (2004). “OsiriX: An Open-Source Software for Navigating in Multidimensional DICOM Images”. English. In: Journal of Digital Imaging 17.3, pp. 205–216 (cit. on pp. 12, 42).

Salakhutdinov, R., A. Mnih, and G. Hinton (2007). “Restricted Boltzmann machines for collaborative filtering”. In: Proceedings of the 24th international conference on Machine learning. ACM, pp. 791–798 (cit. on p. 23).

Schmidhuber, J. (2015). “Deep learning in neural networks: An overview”. In: Neural Networks 61, pp. 85–117 (cit. on p. 22).

Scholl, H. P., M. Fleckenstein, P. C. Issa, C. Keilhauer, F. G. Holz, and B. H. Weber (2007). “An update on the genetics of age-related macular degeneration”. In: Molecular vision 13, p. 196 (cit. on p. 27).

Seddon, J. M., U. A. Ajani, and B. D. Mitchell (1997). “Familial aggregation of age-related maculopathy”. In: American journal of ophthalmology 123.2, pp. 199–206 (cit. on p. 27).

Seddon, J. M., J. Cote, W. F. Page, S. H. Aggen, and M. C. Neale (2005). “The US twin study of age-related macular degeneration: relative roles of genetic and environmental influences”. In: Archives of ophthalmology 123.3, pp. 321–327 (cit. on p. 27).

Seddon, J. M., W. C. Willett, F. E. Speizer, and S. E. Hankinson (1996). “A prospective study of cigarette smoking and age-related macular degeneration in women”. In: Jama 276.14, pp. 1141–1146 (cit. on p. 27).

Smith, W., J. Assink, R. Klein, and P. Mitchell (2001). “Risk factors for age-related macular degeneration: pooled findings from three continents”. In: Ophthalmology (cit. on p. 26).

Smith, W., P. Mitchell, and S. R. Leeder (1996). “Smoking and age-related maculopathy. The Blue Mountains Eye Study.” In: Archives of ophthalmology (Chicago, Ill. : 1960) 114.12, pp. 1518–23 (cit. on p. 26).

Smith, W. and P. Mitchell (1998). “Family history and age-related maculopathy: The Blue Mountains Eye Study”. In: Australian and New Zealand journal of ophthalmology 26.3, pp. 203–206 (cit. on p. 27).

Smolensky, P. (1986). “Chapter 6: Information Processing in Dynamical Systems: Foundations of Harmony Theory.” In: Processing of the Parallel Distributed: Explorations in the Microstructure of Cognition 1 (cit. on p. 22).

Song, H. A. and S.-Y. Lee (2013). “Hierarchical Representation Using NMF”. In: Neural Information Processing. Springer, pp. 466–473 (cit. on p. 22).

Spaide, R. F., H. Koizumi, and M. C. Pozonni (2008). "Enhanced depth imaging spectral-domain optical coherence tomography". In: American journal of ophthalmology 146.4, pp. 496–500 (cit. on p. 38).

Spraul, C. W., G. E. Lang, H. E. Grossniklaus, and G. K. Lang (1999). "Histologic and Morphometric Analysis of the Choroid, Bruch's Membrane, and Retinal Pigment Epithelium in Postmortem Eyes With Age-Related Macular Degeneration and Histologic Examination of Surgically Excised Choroidal Neovascular Membranes". In: Survey of Ophthalmology 44, S10–S32 (cit. on p. 25).

Swanson, E. A., J. Izatt, C. Lin, J. Fujimoto, J. Schuman, M. Hee, D. Huang, and C. Puliafito (1993). "In vivo retinal imaging by optical coherence tomography". In: Optics letters 18.21, pp. 1864–1866 (cit. on p. 28).

Taylor, H. R., G. Tikellis, L. D. Robman, C. A. McCarty, and J. J. McNeil (2002). "Vitamin E supplementation and macular degeneration: randomised controlled trial". In: Bmj 325.7354, p. 11 (cit. on p. 27).

Terrell, G. R. and D. W. Scott (1992). "Variable kernel density estimation". In: The Annals of Statistics, pp. 1236–1265 (cit. on p. 17).

VanNewkirk, M., M. Nanjan, and J. Wang (2000). "The prevalence of age-related maculopathy: The visual impairment project11Received December 30, 1999. Accepted March 27, 2000." In: Ophthalmology (cit. on p. 26).

Velthoven, M. E. van, D. J. Faber, F. D. Verbraak, T. G. van Leeuwen, and M. D. de Smet (2007). "Recent developments in optical coherence tomography for imaging the retina". In: Progress in retinal and eye research 26.1, pp. 57–77 (cit. on p. 34).

Vilnrotter, F. M., R. Nevatia, and K. E. Price (1986). "Structural Analysis of Natural Textures". In: IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-8.1, pp. 76–89 (cit. on p. 14).

Vinding, T., M. Appleyard, J. Nyboe, and G. Jensen (1992). "Risk factor analysis for atrophic and exudative age-related macular degeneration". In: Acta ophthalmologica 70.1, pp. 66–72 (cit. on pp. 26, 27).

Vingerling, J. (1996). "Age-related macular degeneration and smoking: the Rotterdam Study". In: Archives of ... (Cit. on p. 26).

Wiltschi, K., A. Pinz, and T. Lindeberg (2000). "An automatic assessment scheme for steel quality inspection". In: Machine Vision and Applications 12.3, pp. 113–128 (cit. on p. 16).

Witkin, A. J., T. H. Ko, J. G. Fujimoto, A. Chan, W. Drexler, J. S. Schuman, E. Reichel, and J. S. Duker (2006). "Ultra-high resolution optical coherence tomography assessment of photoreceptors in retinitis pigmentosa and related diseases." In: American journal of ophthalmology 142.6, pp. 945–52 (cit. on p. 32).

Xie, X. (2008). A Review of Recent Advances in Surface Defect Detection using Texture analysis Techniques. en (cit. on pp. 5, 6, 14, 15).

Appendices

Source Codes

Note: the following code is co-produced by Dominic Chim and Jing Jing Deng with third party source reference with the code and this report

AMD Feature Extraction

```
clc
clear
close all

pathFolder = 'amd';
d = dir(pathFolder);
isub = [d(:).isdir]; %# returns logical vector
nameFolds = {d(isub).name}';
nameFolds(ismember(nameFolds,{'.','..'})) = [];

u=5; %number of scales
v=8; %number of orientations
gaborArray = gaborFilterBank(u,v,39,39);  % Generates the Gabor filter bank
bing = 0.1;
bins = -1:bing:1;
Feature = [];

for infolder=1:size(nameFolds,1)
    disp(['Processing ' nameFolds{infolder}]);
    files = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.tif'));
    csv = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.csv'));
    Mask = JD_CreateMask([pathFolder '/' nameFolds{infolder} '/' csv.name]);
    Energy = cell(size(files,1),1);
    EnergyMasked = cell(size(files,1),1);
    Feature = cell(size(files,1),1);

    for i=1:size(files,1)
        Feature{i} = zeros(1,numel(1,bins)*u*v);
    end

    for jf=1:size(files,1);
        FileName = [pathFolder '/' nameFolds{infolder} '/' files(jf).name];
        disp(['    Image: ' files(jf).name]);
        img = imread(FileName);  %load our test image
        s=numel(img);
        featureVector = gaborFeatures(img,gaborArray,1,1);   % Extracts Gabor feature vector, 'featureVector',
        c=0;
        result = cell(u,v);
        for i = 1:u
            for j=1:v
                c=c+1;
                result{i,j} = featureVector(((c-1)*s+1):(c*s));
                result{i,j} = reshape(result{i,j},1024,[]);
            end
        end
        Energy{jf} = JD_FeatureEnergy(result, 5, 5, 0.25);
        EnergyMasked{jf} = JD_FeatureMasking(Mask{jf}, Energy{jf});
        result = [];

        for i = 1:size(EnergyMasked{jf},2)
            Feature{jf}(1,(i-1)*numel(bins)+1:i*numel(bins)) = hist(EnergyMasked{jf}(:,i),bins)/size(EnergyMasked{jf},1);
        end
    end

    save([pathFolder '/' nameFolds{infolder} '.Feature.mat'], 'Feature', '-v7.3');
    Energy = [];
    EnergyMasked = [];
end
```

CVNM Feature Extraction

```
clc
clear
close all

pathFolder = 'CVNM';
d = dir(pathFolder);
isub = [d(:).isdir]; %# returns logical vector
nameFolds = {d(isub).name}';
nameFolds(ismember(nameFolds,{'.','..'})) = [];

u=5; %number of scales
v=8; %number of orientations
gaborArray = gaborFilterBank(u,v,39,39);  % Generates the Gabor filter bank
bing = 0.1;
bins = -1:bing:1;
Feature = [];

for infolder=1:size(nameFolds,1)
    disp(['Processing ' nameFolds{infolder}]);
    files = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.tif'));
    csv = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.csv'));
    Mask = JD_CreateMask([pathFolder '/' nameFolds{infolder} '/' csv.name]);
    Energy = cell(size(files,1),1);
    EnergyMasked = cell(size(files,1),1);
    Feature = cell(size(files,1),1);

    for i=1:size(files,1)
        Feature{i} = zeros(1,numel(1,bins)*u*v);
    end

    parfor jf=1:size(files,1);
        FileName = [pathFolder '/' nameFolds{infolder} '/' files(jf).name];
        disp(['    Image: ' files(jf).name]);
        img = imread(FileName);  %load our test image
        s=numel(img);
        featureVector = gaborFeatures(img,gaborArray,1,1);   % Extracts Gabor feature vector, '
   featureVector',
        c=0;
        result = cell(u,v);
        for i = 1:u
            for j=1:v
                c=c+1;
                result{i,j} = featureVector(((c-1)*s+1):(c*s));
                result{i,j} = reshape(result{i,j},1024,[]);
            end
        end
        Energy{jf} = JD_FeatureEnergy(result, 5, 5, 0.25);
        EnergyMasked{jf} = JD_FeatureMasking(Mask{jf}, Energy{jf});
        result = [];

        for i = 1:size(EnergyMasked{jf},2)
            Feature{jf}(1,(i-1)*numel(bins)+1:i*numel(bins)) = hist(EnergyMasked{jf}(:,i),bins)/
   size(EnergyMasked{jf},1);
```

```
        end
    end

    save([pathFolder '/' nameFolds{infolder} '.Feature.mat'], 'Feature', '-v7.3');
    Energy = [];
    EnergyMasked = [];
end
```

PED Feature Extraction

```
clc
clear
close all

pathFolder = 'PED';
d = dir(pathFolder);
isub = [d(:).isdir]; %# returns logical vector
nameFolds = {d(isub).name}';
nameFolds(ismember(nameFolds,{'.','..'})) = [];

u=5; %number of scales
v=8; %number of orientations
gaborArray = gaborFilterBank(u,v,39,39);  % Generates the Gabor filter bank
bing = 0.1;
bins = -1:bing:1;
Feature = [];

for infolder=1:size(nameFolds,1)
    disp(['Processing ' nameFolds{infolder}]);
    files = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.tif'));
    csv = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.csv'));
    Mask = JD_CreateMask([pathFolder '/' nameFolds{infolder} '/' csv.name]);
    Energy = cell(size(files,1),1);
    EnergyMasked = cell(size(files,1),1);
    Feature = cell(size(files,1),1);

    for i=1:size(files,1)
        Feature{i} = zeros(1,numel(1,bins)*u*v);
    end

    for jf=1:size(files,1);
        FileName = [pathFolder '/' nameFolds{infolder} '/' files(jf).name];
        disp(['    Image: ' files(jf).name]);
        img = imread(FileName);  %load our test image
        s=numel(img);
        featureVector = gaborFeatures(img,gaborArray,1,1);   % Extracts Gabor feature vector, 'featureVector',
        c=0;
        result = cell(u,v);
        for i = 1:u
            for j=1:v
                c=c+1;
                result{i,j} = featureVector(((c-1)*s+1):(c*s));
                result{i,j} = reshape(result{i,j},1024,[]);
            end
        end
        Energy{jf} = JD_FeatureEnergy(result, 5, 5, 0.25);
        EnergyMasked{jf} = JD_FeatureMasking(Mask{jf}, Energy{jf});
        result = [];

        for i = 1:size(EnergyMasked{jf},2)
```

```
            Feature{jf}(1,(i-1)*numel(bins)+1:i*numel(bins)) = hist(EnergyMasked{jf}(:,i),bins)/size(EnergyMasked{jf},1);
        end
    end

    save([pathFolder '/' nameFolds{infolder} '.Feature.mat'], 'Feature', '-v7.3');
    Energy = [];
    EnergyMasked = [];
end
```

Norm Feature Extraction

```
clc
clear
close all

pathFolder = 'Norm';
d = dir(pathFolder);
isub = [d(:).isdir]; %# returns logical vector
nameFolds = {d(isub).name}';
nameFolds(ismember(nameFolds,{'.','..'})) = [];

u=5; %number of scales
v=8; %number of orientations
gaborArray = gaborFilterBank(u,v,39,39);  % Generates the Gabor filter bank
bing = 0.1;
bins = -1:bing:1;
Feature = [];

for infolder=1:size(nameFolds,1)
    disp(['Processing ' nameFolds{infolder}]);
    files = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.tif'));
    csv = dir(fullfile([pathFolder '/' nameFolds{infolder} '/'],filesep,'*.csv'));
    Mask = JD_CreateMask([pathFolder '/' nameFolds{infolder} '/' csv.name]);
    Energy = cell(size(files,1),1);
    EnergyMasked = cell(size(files,1),1);
    Feature = cell(size(files,1),1);

    for i=1:size(files,1)
        Feature{i} = zeros(1,numel(1,bins)*u*v);
    end

    for jf=1:size(files,1);
        FileName = [pathFolder '/' nameFolds{infolder} '/' files(jf).name];
        disp(['    Image: ' files(jf).name]);
        img = imread(FileName);  %load our test image
        s=numel(img);
        featureVector = gaborFeatures(img,gaborArray,1,1);   % Extracts Gabor feature vector, 'featureVector',
        c=0;
        result = cell(u,v);
        for i = 1:u
            for j=1:v
                c=c+1;
                result{i,j} = featureVector(((c-1)*s+1):(c*s));
                result{i,j} = reshape(result{i,j},1024,[]);
            end
        end
        Energy{jf} = JD_FeatureEnergy(result, 5, 5, 0.25);
        EnergyMasked{jf} = JD_FeatureMasking(Mask{jf}, Energy{jf});
        result = [];

        for i = 1:size(EnergyMasked{jf},2)
            Feature{jf}(1,(i-1)*numel(bins)+1:i*numel(bins)) = hist(EnergyMasked{jf}(:,i),bins)/size(EnergyMasked{jf},1);
        end
    end

    save([pathFolder '/' nameFolds{infolder} '.Feature.mat'], 'Feature', '-v7.3');
    Energy = [];
    EnergyMasked = [];
end
```

Test Data Assembler

```
clc;
clear;
close all;

Classes = {{'Norm'}; {'CVNM'}; {'AMD'}; {'PED'}};
Data = cell(size(Classes,1),1);

for ic = 1:size(Classes,1)
    matfiles = dir(fullfile([Classes{ic}{1} '/'],filesep,'*.mat'));
    PA = cell(size(matfiles,1),1);
    for im = 1:size(matfiles,1)
        Feature = [];
        load(fullfile([Classes{ic}{1} '/' matfiles(im).name]));
        PA{im} = Feature;
    end
    Data{ic} = PA;
end

save('Data', 'Data', '-v7.3');
```

Confusion Matrix

```
function [Confusion, NConfusion] = JD_ConfusionMatrix(Prediction, GroundTurth, nClasses)
%JD_CONFUSIONMATRIX Summary of this function goes here
%   Detailed explanation goes here
% Col ground turth, Row Prediction
Confusion = zeros(nClasses, nClasses);
NConfusion = zeros(nClasses, nClasses);
Normal = hist(GroundTurth,1:nClasses);

for i=1:nClasses %Prediction
    SelPred = find(Prediction == i);
    GouT = GroundTurth(SelPred);
    for j=1:nClasses %Ground Turth
        SelGouT = find(GouT == j);
        Confusion(i,j) = numel(SelGouT);
        NConfusion(i,j) = Confusion(i,j)/Normal(j);
    end
end

end
```

Cross Fold Validation

```
clc
clear
close all;

addpath('./SVM');

load('Data.mat');
```

```
NumRun = 10;
NumFolder = 10;
NumClass = size(Data,1);

% Prepare the data for multiple Run
DataNumFolder = cell(size(Data,1),1);
for i=1:size(DataNumFolder,1)
    DT = Data{i};
    for j=1:size(DT,1)
        for k=1:size(Data{i}{j},1);
            DataNumFolder{i} = [DataNumFolder{i}; Data{i}{j}{k}];
        end
    end
end

DataMultiRun = cell(NumRun,1);
for R = 1:NumRun
    % Random the samples
    DataNumFolderRun = cell(size(Data,1),1);

    for i=1:size(DataNumFolder,1)
        RandSel = randperm(size(DataNumFolder{i},1));
        DataNumFolderRun{i} = DataNumFolder{i}(RandSel,:);
    end

    % Separate NumFolder
    DataNumFolderTrn = cell(NumFolder,size(Data,1));
    DataNumFolderTst = cell(NumFolder,size(Data,1));
    for C = 1:size(Data,1)
        NumFolderSize = floor(size(DataNumFolderRun{i},1)/NumFolder);
        for F = 1:NumFolder
            TestSel = [];
            if(F~=NumFolder)
                TstSel = (F-1)*NumFolderSize+1:F*NumFolderSize;
            else
                TstSel = (F-1)*NumFolderSize+1:size(DataNumFolderRun{i},1);
            end
            TrnSel = 1:size(DataNumFolderRun{i},1);
            TrnSel(TstSel) = [];
            DataNumFolderTst{F,C} = DataNumFolderRun{C}(TstSel,:);
            DataNumFolderTrn{F,C} = DataNumFolderRun{C}(TrnSel,:);
        end
    end
    DataMultiRun{R}.Trn = DataNumFolderTrn;
    DataMultiRun{R}.Tst = DataNumFolderTst;
end

% Run
ResultKNN = cell(NumRun, NumFolder);
ResultRFC = cell(NumRun, NumFolder);
ResultSVM = cell(NumRun, NumFolder);
NResultKNN = cell(NumRun, NumFolder);
NResultRFC = cell(NumRun, NumFolder);
NResultSVM = cell(NumRun, NumFolder);
ModelKNN = cell(NumRun, NumFolder);
ModelRFC = cell(NumRun, NumFolder);
ModelSVM = cell(NumRun, NumFolder);

for R = 1:NumRun
    disp(['Run:' num2str(R)]);
    DataNumFolderTrn =  DataMultiRun{R}.Trn;
    DataNumFolderTst =  DataMultiRun{R}.Tst;

    for F = 1:NumFolder
        disp(['    NumFolder:' num2str(F)]);
        TrnData = [];
        TrnClas = [];
        TstData = [];
        TstClas = [];
        for C = 1:size(Data,1)
            TrnData = [TrnData; DataNumFolderTrn{F,C}];
            TrnClas = [TrnClas; C*ones(size(DataNumFolderTrn{F,C},1),1)];
            TstData = [TstData; DataNumFolderTst{F,C}];
            TstClas = [TstClas; C*ones(size(DataNumFolderTst{F,C},1),1)];
        end

%          % Train & Test with K-NN
%          ModelKNN{R,F} = fitcknn(TrnData, TrnClas, 'NumNeighbors', 5, 'Distance', 'euclidean');
%          PDT = predict(ModelKNN{R,F}, TstData);
%          [CF,NC] = JD_ConfusionMatrix(PDT, TstClas, numClass);
%          ResultKNN{R,F} = CF;
%          NResultKNN{R,F} = NC;
%
%          % Train & Test with RFC
%          ModelRFC{R,F} = TreeBagger(50, TrnData, TrnClas, 'Method', 'classification', 'MinLeaf', 3);
%          PDT = str2num(cell2mat(predict(ModelRFC{R,F}, TstData)));
%          [CF,NC] = JD_ConfusionMatrix(PDT, TstClas, numClass);
%          ResultRFC{R,F} = CF;
%          NResultRFC{R,F} = NC;
%
%          % Train & Test with SVM
%          ModelSVM{R,F} = svmtrain(TrnClas, TrnData, ['-s 0 -t 2 -g ' num2str(2^(3)) ' -c ' num2str(2^(5))]);
%          [PDT, acc] = svmpredict(TstClas, TstData, ModelSVM{R,F});
%          [CF,NC] = JD_ConfusionMatrix(PDT, TstClas, numClass);
%          ResultSVM{R,F} = CF;
%          NResultSVM{R,F} = NC;
%
        %% Train RBM for classification
%train rbm with 100 hidden units
m=rbmFit(TrnData,50,TrnClas,'verbose',true);
yhat=rbmPredict(m,TstData);

%print error
fprintf('Classification error using RBM with 100 hiddens is %f\n', ...
    sum(yhat~= TstClas)/length(yhat));
ResultRBM{R,F}=sum(yhat~= TstClas)/length(yhat);



%% Train a DBN
op.verbose=true;
models=dbnFit(TrnData,[100 100],TrnClas,op,op);
yhat2=dbnPredict(models,TstData);

%print error
fprintf('Classification error using DBN with 100-100 hiddens is %f\n', ...
    sum(yhat2~= TstClas)/length(yhat2));
ResultDBN{R,F}=sum(yhat~= TstClas)/length(yhat);


```

```


    end
end

save('DataNumFolder.EXP02.mat', '-v7.3');
```

Leave-one-out

```
clc
clear
close all;

addpath('./SVM');

load('Data.mat');
NumRun = 10;
NumPat = 7;
NumClass = size(Data,1);

% Prepare the data for multiple Run
DataMultiRun = cell(NumRun,1);
for R=1:NumRun
    TrnSet = cell(NumPat, NumClass);
    TstSet = cell(NumPat, NumClass);
    for ip=1:NumPat
        for ic=1:NumClass
            Sel = randperm(7);
            DT = Data{ic}(Sel);
            TstSetT = [];
            for i=1:size(DT{1},1)
                TstSetT = [TstSetT; DT{1}{i}];
            end

            TrnSetT = [];
            for j=2:NumPat
                for i=1:size(DT{j},1)
                    TrnSetT = [TrnSetT; DT{j}{i}];
                end
            end
            TrnSet{ip,ic} = TrnSetT;
            TstSet{ip,ic} = TstSetT;
        end
    end
    DataMultiRun{R}.Trn = TrnSet;
    DataMultiRun{R}.Tst = TstSet;
end

% Run
ResultKNN = cell(NumRun, NumPat);
ResultRFC = cell(NumRun, NumPat);
ResultSVM = cell(NumRun, NumPat);
NResultKNN = cell(NumRun, NumPat);
NResultRFC = cell(NumRun, NumPat);
NResultSVM = cell(NumRun, NumPat);
ModelKNN = cell(NumRun, NumPat);
ModelRFC = cell(NumRun, NumPat);
ModelSVM = cell(NumRun, NumPat);

for R = 1:NumRun
    disp(['Run:' num2str(R)]);
    DataFolderTrn =  DataMultiRun{R}.Trn;
    DataFolderTst =  DataMultiRun{R}.Tst;

    for F = 1:NumPat
        disp(['    Pat:' num2str(F)]);
        TrnData = [];
        TrnClas = [];
        TstData = [];
        TstClas = [];
        for C = 1:size(Data,1)
            TrnData = [TrnData; DataFolderTrn{F,C}];
            TrnClas = [TrnClas; C*ones(size(DataFolderTrn{F,C},1),1)];
            TstData = [TstData; DataFolderTst{F,C}];
            TstClas = [TstClas; C*ones(size(DataFolderTst{F,C},1),1)];
        end

%         % Train & Test with K-NN
%         ModelKNN{R,F} = fitcknn(TrnData, TrnClas, 'NumNeighbors', 5, 'Distance', 'euclidean');
%         PDT = predict(ModelKNN{R,F}, TstData);
%         [CF,NC] = JD_ConfusionMatrix(PDT, TstClas, NumClass);
%         ResultKNN{R,F} = CF;
%         NResultKNN{R,F} = NC;
%
%         % Train & Test with RFC
%         ModelRFC{R,F} = TreeBagger(50, TrnData, TrnClas, 'Method', 'classification', 'MinLeaf',
     3);
%         PDT = str2num(cell2mat(predict(ModelRFC{R,F}, TstData)));
%         [CF,NC] = JD_ConfusionMatrix(PDT, TstClas, NumClass);
%         ResultRFC{R,F} = CF;
%         NResultRFC{R,F} = NC;
%
%         % Train & Test with SVM
%         ModelSVM{R,F} = svmtrain(TrnClas, TrnData, ['-s 0 -t 2 -g ' num2str(2^(3)) ' -c '
    num2str(2^(5))]);
%         [PDT, acc] = svmpredict(TstClas, TstData, ModelSVM{R,F});
%         [CF,NC] = JD_ConfusionMatrix(PDT, TstClas, NumClass);
%         ResultSVM{R,F} = CF;
%         NResultSVM{R,F} = NC;

        %% Train RBM for classification
%train rbm with 100 hidden units
m=rbmFit(TrnData,50,TrnClas,'verbose',true);
yhat=rbmPredict(m,TstData);

%print error
fprintf('Classification error using RBM with 100 hiddens is %f\n', ...
    sum(yhat~= TstClas)/length(yhat));
ResultRBM{R,F}=sum(yhat~= TstClas)/length(yhat);



%% Train a DBN
op.verbose=true;
models=dbnFit(TrnData,[100 100],TrnClas,op,op);
yhat2=dbnPredict(models,TstData);

%print error
fprintf('Classification error using DBN with 100-100 hiddens is %f\n', ...
    sum(yhat2~= TstClas)/length(yhat2));
ResultDBN{R,F}=sum(yhat~= TstClas)/length(yhat);

```

```



    end
end

save('DataFolder.EXPPA.mat', '-v7.3');
```

Result Analysis

```
clc
clear
close all
%%
load('DataFolder.EXPPA.mat');
if(~exist('numClass','var') && exist('NumClass','var'))
    numClass = NumClass;
end

if(~exist('Folder','var') && exist('NumPat','var'))
    Folder = NumPat;
end

NCKNNALL = [];
NCRFCALL = [];
NCSVMALL = [];
for R=1:NumRun
    for F=1:Folder
        NCKNNALL = [NCKNNALL; reshape(NResultKNN{R,F},1,numClass*numClass)];
        NCRFCALL = [NCRFCALL; reshape(NResultRFC{R,F},1,numClass*numClass)];
        NCSVMALL = [NCSVMALL; reshape(NResultSVM{R,F},1,numClass*numClass)];
    end
end

NCKNNMean = reshape(mean(NCKNNALL,1), numClass, numClass);
NCRFCMean = reshape(mean(NCRFCALL,1), numClass, numClass);
NCSVMMean = reshape(mean(NCSVMALL,1), numClass, numClass);

NCKNNSTD = reshape(std(NCKNNALL,1), numClass, numClass);
NCRFCSTD = reshape(std(NCRFCALL,1), numClass, numClass);
NCSVMSTD = reshape(std(NCSVMALL,1), numClass, numClass);
```

Create mask

```
function [ bw ] = JD_CreateMask(csvfile)
%JD_MASKFEATURE Summary of this function goes here
%   Detailed explanation goes here
[headers Data Name FrameNum parsedContent] = parseOsirixROIfile(csvfile);
sizeData=length(Data)-1;
bw = cell(sizeData,1);
for i = 1:sizeData
    pts= Data{i};
    pts=[pts,pts(:,1)];
    Y = interppolygon((pts'),1500);
    bw{i} = poly2mask(Y(:,1),Y(:,2),1024,512);
end
end
```

Parse Osirix ROI

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This Function takes a CSV (comma separated value) file as an input specified in :
% - filename
% and returns:
% - headers: the name of the columns in the original CSV file
% - Data: a Cell array of the [x,y] coordinates of key points of the
%   contour
% - Name: The name of the contour (Adventitia, Lumen,...)
% - FrameNum: the frame index of the IVUS image
% - parsedContent: the original data
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Written by C. Charron October 2010


function [headers Data Name FrameNum parsedContent] = parseOsirixROIfile(filename)
content = textread(filename,'%s','delimiter','');

% first, read the headers
tmp = content{1};
pos = find( tmp==',' );
pos = [0 pos length(tmp)+1];
for i=2:length(pos)
    headers{i-1} = tmp(pos(i-1)+1:pos(i)-1);
end

% find the type of region associated to each ROI
for i=1:length(headers)
    switch headers{i}
        case 'RoiName'
            borderTypeColumn = i;
        case 'NumOfPoints'
            numOfPointsColumn = i;
        case 'mmX'
            dataStartColumn = i;
        case 'ImageNo'
            imageNumColumn = i;
        otherwise
    end
end

% parse each line of the content
for i=2:length(content)
    tmp = content{i};
    pos = find( tmp==',' );
    pos = [0 pos length(tmp)+1];
    for j=2:length(pos)
        parsedContent(i-1).line{j-1} = tmp(pos(j-1)+1:pos(j)-1);
    end
end


% get the data for each line. Organised as a cell
for i=1:length(parsedContent)
    Name{i} = parsedContent(i).line{borderTypeColumn};
    FrameNum{i} = str2num(parsedContent(i).line{imageNumColumn});
    Data{i} = [];
    for j=dataStartColumn:length(parsedContent(i).line)
        Data{i} = [Data{i} str2num(parsedContent(i).line{j})];
    end
    length(Data(i))/5;
    Data{i} = reshape(Data{i},[5 length(Data{i})/5]);
    Data{i} = Data{i}(4:5,:);
    pos = find((Data{i}(1,:)|Data{i}(2,:))==0);
    Data{i}(1:2,pos)=[];
end
```

Polygon creation

```
function varargout = interppolygon(X,N,method)
% INTERPPOLYGON  Interpolates a polygon.
%
%   Y = INTERPPOLYGON(X,N) interporpolates a polygon whose coordinates are
%   given in X over N points, and returns it in Y.
%
%   The polygon X can be of any dimension (2D, 3D, ...), and coordinates
%   are exected to be along columns. For instance, if X is 10x2 matrix (2D
%   polygon of 10 points), then Y = INTERPPOLYGON(X,50) will return a
%   50x2 matrix.
%
%   INTERPPOLYGON uses interp1 for interpolation. The interpolation method
%   can be specified as 3rd argument, as in Y=INTERPPOLYGON(X,N,'method').
%   Allowed methods are 'nearest', 'linear', 'spline' (the default),
%   'pchip', 'cubic' (see interp1).
%
%   ALGORITHM
%   The point to point distance array is used to build a metric, which will
%   be interpolated by interp1 over the given number of points.
%   Interpolated points are thus equally spaced only in the linear case,
%   not in other cases where interpolated points do not lie on the initial
%   polygon. If this is an issue, try caalling INTERPPOLYGON twice, as in:
%   >> Y = INTERPPOLYGON(X,50); % Make a spline interpolation%
%   >> Z = INTERPPOLYGON(Y,50); % Will correct for uneven space between points
%
%   OUTPUT
%   On top of the interpolated polygon, [Y M] = INTERPPOLYGON(X,N) will
%   return in M the metric of the original polygon.
%
%   EXAMPLE
%   X = rand(5,2);
%   Y = interppolygon(X,50);
%   figure, hold on
%   plot(X(:,1),X(:,2),'ko-')
%   plot(Y(:,1),Y(:,2),'r.-')

% my example
% Y = interppolygon(round(Data{1,1}'),1507);
% gg=zeros(480,480);
% gg(sub2ind(size(gg),round(Y(:,2)),round(Y(:,1))))=1;
% figure;imagesc(gg)

% -------------------- INFO --------------------
%   Authors: Jean-Yves Tinevez
%   Work address: Max-Plank Institute for Cell Biology and Genetics,
%   Dresden,  Germany.
%   Email: tinevez AT mpi-cbg DOT de
%   January 2009
%   Permission is given to distribute and modify this file as long as this
%   notice remains in it. Permission is also given to write to the author
%   for any suggestion, comment, modification or usage.
% -------------------- BEGIN CODE --------------------

    if nargin < 3
        method = 'PCHIP';
    end
    if nargin < 2 || N < 2
```

```
        N = 2;
    end

    % 1. Build metric
    [pl bl]         = polygonlength(X);
    orig_metric     = [   0 ;  cumsum(bl/pl) ];

    % 2. Interpolate
    interp_metric   = ( 0 : 1/(N-1) : 1)';
    Y               = interp1( ...
        orig_metric,...
        X,...
        interp_metric,...
        method);

    % 3. Ouputs
    varargout{1} = Y;
    if nargout > 1
        varargout{2} = orig_metric;
    end

    %% Subfunction
    % Calculate the point to point distanceof each polygon point
    function varargout = polygonlength(X)

        n_dim       = size(X,2);
        delta_X     = 0;
        for dim = 1 : n_dim
            delta_X = delta_X + ...
                diff(X(:,dim)).^2 ;
        end

        branch_lengths  = sqrt( delta_X );
        pl              = sum( branch_lengths );

        varargout{1}    = pl;
        if nargout > 1
            varargout{2} = branch_lengths;
        end

    end

end
```

Gabor Filter Bank

```
function gaborArray = gaborFilterBank(u,v,m,n)

% GABORFILTERBANK generates a custum Gabor filter bank.
% It creates a u by v array, whose elements are m by n matries;
% each matrix being a 2-D Gabor filter.
%
%
% Inputs:
%       u : No. of scales (usually set to 5)
%       v : No. of orientations (usually set to 8)
%       m : No. of rows in a 2-D Gabor filter (an odd integer number usually set to 39)
%       n : No. of columns in a 2-D Gabor filter (an odd integer number usually set to 39)
%
% Output:
%       gaborArray: A u by v array, element of which are m by n
%                   matries; each matrix being a 2-D Gabor filter
```

```
%
%
% Sample use:
%
% gaborArray = gaborFilterBank(5,8,39,39);
%
%
%   Details can be found in:
%
%  M. Haghighat, S. Zonouz, M. Abdel-Mottaleb, "Identification Using
%   Encrypted Biometrics," Computer Analysis of Images and Patterns,
%   Springer Berlin Heidelberg, pp. 440-448, 2013.
%
%
% (C) Mohammad Haghighat, University of Miami
%       haghighat@ieee.org
%       I WILL APPRECIATE IF YOU CITE OUR PAPER IN YOUR WORK.



if (nargin ~= 4)    % Check correct number of arguments
    error('There should be four inputs.')
end


%% Create Gabor filters

% Create u*v gabor filters each being an m*n matrix

gaborArray = cell(u,v);
fmax = 0.25;
gama = sqrt(2);
eta = sqrt(2);

for i = 1:u

    fu = fmax/((sqrt(2))^(i-1));
    alpha = fu/gama;
    beta = fu/eta;

    for j = 1:v
        tetav = ((j-1)/v)*pi;
        gFilter = zeros(m,n);

        for x = 1:m
            for y = 1:n
                xprime = (x-((m+1)/2))*cos(tetav)+(y-((n+1)/2))*sin(tetav);
                yprime = -(x-((m+1)/2))*sin(tetav)+(y-((n+1)/2))*cos(tetav);
                gFilter(x,y) = (fu^2/(pi*gama*eta))*exp(-((alpha^2)*(xprime^2)+(beta^2)*(yprime^2)))*exp(1i*2*pi*fu*xprime);
            end
        end
        gaborArray{i,j} = gFilter;

    end
end


%% Show Gabor filters

% Show magnitudes of Gabor filters:
```

```
% figure('NumberTitle','Off','Name','Magnitudes of Gabor filters');
% for i = 1:u
%     for j = 1:v
%         subplot(u,v,(i-1)*v+j);
%         imshow(abs(gaborArray{i,j}),[]);
%     end
% end
%
% % Show real parts of Gabor filters:
% figure('NumberTitle','Off','Name','Real parts of Gabor filters');
% for i = 1:u
%     for j = 1:v
%         subplot(u,v,(i-1)*v+j);
%         imshow(real(gaborArray{i,j}),[]);
%     end
% end
```

Gabor Filter

```
function featureVector = gaborFeatures(img,gaborArray,d1,d2)

% GABORFEATURES extracts the Gabor features of the image.
% It creates a column vector, consisting of the image's Gabor features.
% The feature vectors are normalized to zero mean and unit variance.
%
%
% Inputs:
%       img         : Matrix of the input image
%       gaborArray  : Gabor filters bank created by the function gaborFilterBank
%       d1          : The factor of downsampling along rows.
%                       d1 must be a factor of n if n is the number of rows in img.
%       d2          : The factor of downsampling along columns.
%                       d2 must be a factor of m if m is the number of columns in img.
%
% Output:
%       featureVector :   A column vector with length (m*n*u*v)/(d1*d2).
%                           This vector is the Gabor feature vector of an
%                           m by n image. u is the number of scales and
%                           v is the number of orientations in 'gaborArray'.
%
%
% Sample use:
%
% img = imread('cameraman.tif');
% gaborArray = gaborFilterBank(5,8,39,39);  % Generates the Gabor filter bank
% featureVector = gaborFeatures(img,gaborArray,4,4);   % Extracts Gabor feature vector, 'featureVector', from the image, 'img'.
%
%
%   Details can be found in:
%
%   M. Haghighat, S. Zonouz, M. Abdel-Mottaleb, "Identification Using
%   Encrypted Biometrics," Computer Analysis of Images and Patterns,
%   Springer Berlin Heidelberg, pp. 440-448, 2013.
%
%
% (C) Mohammad Haghighat, University of Miami
%       haghighat@ieee.org
%       I WILL APPRECIATE IF YOU CITE OUR PAPER IN YOUR WORK.


if (nargin ~= 4)    % Check correct number of arguments
```

```
    error('Use correct number of input arguments!')
end

if size(img,3) == 3 % % Check if the input image is grayscale
    img = rgb2gray(img);
end

img = double(img);


%% Filtering

% Filter input image by each Gabor filter
[u,v] = size(gaborArray);
gaborResult = cell(u,v);
for i = 1:u
    for j = 1:v
        gaborResult{i,j} = conv2(img,gaborArray{i,j},'same');
        % J{u,v} = filter2(G{u,v},I);
    end
end


%% Feature Extraction

% Extract feature vector from input image
[n,m] = size(img);
s = (n*m)/(d1*d2);
l = s*u*v;
featureVector = zeros(l,1);
c = 0;
for i = 1:u
    for j = 1:v

        c = c+1;
        gaborAbs = abs(gaborResult{i,j});
        gaborAbs = downsample(gaborAbs,d1);
        gaborAbs = downsample(gaborAbs.',d2);
        gaborAbs = reshape(gaborAbs.',[],1);

        % Normalized to zero mean and unit variance. (if not applicable, please comment this line)
        gaborAbs = (gaborAbs-mean(gaborAbs))/std(gaborAbs,1);

        featureVector(((c-1)*s+1):(c*s)) = gaborAbs;

    end
end


%% Show filtered images

% % Show real parts of Gabor-filtered images
% figure('NumberTitle','Off','Name','Real parts of Gabor filters');
% for i = 1:u
%     for j = 1:v
%         subplot(u,v,(i-1)*v+j)
%         imshow(real(gaborResult{i,j}),[]);
%     end
% end
%
```

```
%
% % Show magnitudes of Gabor-filtered images
% figure('NumberTitle','Off','Name','Magnitudes of Gabor filters');
% for i = 1:u
%     for j = 1:v
%         subplot(u,v,(i-1)*v+j)
%         imshow(abs(gaborResult{i,j}),[]);
%     end
% end
```

Energy Transform

```
function [ energy ] = JD_FeatureEnergy(input, w, h, alpha)
%JD_FEATUREENERGY Summary of this function goes here
%   Detailed explanation goes here
kernel = ones(w,h);
[m,n] = size(input);
energy = cell(m,n);

for i = 1:m
    for j = 1:n
        I = input{i,j};
        E = exp(-2*alpha*I);
        T = abs((1-E)./(1+E));
        energy{i,j} = imfilter(T, kernel,'replicate')/(w*h);
    end
end

end
```

Feature Masking

```
function [ masked ] = JD_FeatureMasking( mask, energy )
%JD_FEATUREMASKING Summary of this function goes here
%   Detailed explanation goes here
[u,v] = size(energy);
select = find(mask==1);
masked = zeros(numel(select),u*v);

for i=1:u
    for j=1:v
        se = energy{i,j}(select);
        masked(:,v*(i-1)+j) = se;
    end
end

end
```