Deep Learning Methods for Texture Analysis in Medical Imaging

Dafydd Ravenscroft

847620

Swansea University
Prifysgol Abertawe

Department of Computer Science

Swansea University

supervised by

Dr. Xianghua Xie

January 2018

Summary

Deep learning is a rapidly growing area of research due to the strong results it is able to produce, particularly in areas of image recognition, object detection and semantic segmentation. There have been some recent works exploring its use in the field of texture analysis. In this work we explore deep learning techniques applied to texture classification using a medical imaging dataset. We particularly deal with a dataset with irregularly sized and shaped input images, developing novel methods to deal with this challenge.

Image classification can be applied in a medical setting to categorise medical images into stages or types of disease. In this work we look at using deep learning for the classification of stages of Age-Related Macular Degeneration, a progressive eye disease which leads to vision loss. The disease is particularly prevalent amongst the elderly and if diagnosed in its early stages can be treated simply to prevent sight loss developing. We have a dataset of 75 patients' eyes split into three equally sized categories: healthy, early and wet. We concentrate on using the choroidal layer of the eye for classification, performing texture analysis on it.

We propose novel deep learning techniques to perform texture classification on the inconsistently shaped and sized dataset. We firstly use an approach in which a set of regularly shaped patches are extracted from each of the input images. The feature learning abilities of convolutional neural networks is utilised to train feature extractors to learn the textural information contained in the choroidal images. Having used these feature extractors to develop a set of low-level features we introduce a couple methods to develop higher level features and for generalisation. Machine learning classifiers are then used to categorise inputs into one of the three classes. We demonstrate these methods produce accurate results for 2-fold and 10-fold cross validation on the AMD dataset which outperform hand-crafted feature extraction techniques.

We secondly introduce an original approach for being able to take irregular shaped images as input into a convolutional network. This is achieved by the use of a set of conventional layers to produce a learnable histogram within the framework and by using masking layers to ensure only the region of interest is used. We test this approach on 2-fold and leave-one-patient-out cross validation demonstrating the potential of this method.

Declaration

This work has not previously been accepted in substance for any degree and is not being concurrently submitted in candidature for any degree.

STATEMENT 1

This thesis is the result of my own independent work/investigation, except where otherwise stated. Other sources are acknowledged by giving explicit references. A bibliography is appended.

Signed ....

Date ....

STATEMENT 2

I hereby give consent for my thesis, if accepted, to be available for photocopying and for inter-library loan, and for the title and summary to be made available to outside organisations.

Signed ....

Date ....

Contents

Chapter 1

Introduction

In this section the problems we aim to undertake are introduced. The reasons for the choice of this project is explained and what we aim to achieve from it. We provide a concise overview of the motivation and methodology for this research.

1.1 Motivation

Deep learning provides powerful tools which are widely used in computer vision problems such as object recognition and image segmentation. Texture analysis can be important for the classification of different objects and this is often the case in medical imaging. However, the application of deep learning for texture analysis applied in this field is relatively underrepresented. Due to the success deep learning presents in other similar areas it is logical to assume it too will be in this area.

We are also particularly interested in developing deep learning methods which work with datasets in which the size and shape of the input images is not uniform. A requisite of most deep learning techniques is that inputs are all of the consistent dimensions. Overcoming this necessity would allow inputs of any size to all be trained in a single network.

We will concentrate particularly on one specific application, the diagnosis of stages of Age-Related Macular Degeneration (AMD). This is a progressive eye disease which causes damage to the retina and can lead to vision loss. It is the leading cause of vision loss in the Western world, particularly among the elderly and is becoming increasingly prevalent due to aging populations [1, 2]. When detected in time, treatment is simple and highly effective at halting spread of the disease highlighting the importance of timely detection. Traditionally AMD is diagnosed by qualified optometrists examining retinal shape and appearance. Recent technological advances allow more accurate images of deep sections, such as the choroid, of the eye. This area is known to be affected by AMD and its appearance changed [3]. Using only the segmented choroid of patients' eyes provides us with a set of images which all vary in size and shape. Due to the newness of the imaging technique little study has been conducted into using it for classification particularly using machine learning and deep learning techniques.

1.2 Overview

In this project we explore using machine learning and deep learning techniques for texture recognition problems, notably in the field of medical imaging. We investigate a number of different methods for textural-based classification. We concentrate out attention in particular on one AMD dataset which is used as a basis to compare the effectiveness and accuracy of our differing methodologies. The dataset presents images of varying sizes and shapes from sample to sample, making applying conventional techniques directly to the data difficult, and this precipitates our chose of methodologies. We follow two main approaches, although different methods are analysed within these:

- Patch based - Here, we extract square patches from the regions of interest from the dataset and use these for training our convolutional networks and classifiers. The challenges presented from this technique are that the patches are relatively small compared to the whole image. Further, each patch from an image has the same classification as its parent image but may not necessarily contain elements that demonstrate that class as defining features may be localised to certain areas of the image. We aim for a fully automated approach for learning features. Our methods utilise using convolutional networks to automatically learn feature extractors before using a number of stages to develop higher-level features which then pass onto classifiers for categorisation.

- Whole image - Convolutional networks require inputs to be of consistent size and shape. To overcome this problem we present a novel technique in which a histogram layer and a masking layer are introduced into the framework of a convolutional network. This allows inputs of any dimensions to be compatible and the network to learn classifications for them. We combine this with a ResNet architecture in an end-to-end architecture.

Deep learning is a powerful tool and in this project we present methods for utilising it to overcome tangible problems in medical imaging. We also offer novel approaches for overcoming problems presented by irregular datasets. The deep learning techniques proposed present positive results and lend a strong basis for further expansion of these methods.

1.3 Thesis Layout

The rest of this thesis is organised as follows:

- **Chapter 2** - *Background:* This chapter provides an outline of the development of the area of deep learning and looks at the current state-of-the-art methods. We also examine the area of texture analysis, looking at traditional and deep learning approaches.

- **Chapter 3** - *Dataset:* In this chapter we cover the dataset, providing an outline of what AMD is and the ophthalmological method for obtaining the data we are to use. It also provides a summary of related works, including representatives of AMD detection and studies concentrating on the choroid.

- **Chapter 4** - *Patch-based approach:* In this chapter we discuss our method of extracting patches from the choroidal region to develop our classifiers and introduce a number of methods using the data in this form.

- **Chapter 5** - *End-to-end histogram framework:* Here we propose our novel method which enables the use of the whole segmented choroidal regions by introducing histograms into the framework of a convolutional network.

- **Chapter 6** - *Conclusions and future work:* This chapter looks at the research findings and analyses the success of the methodology. We discuss potential areas for improvement and extensions which could be made to the project.

1.4 List of publications

The following is a list of published papers as a result of this work:

1. Ravenscroft D, Deng J, Xie X, Terry L, Margrain TH, North RV, Wood A. Learning feature extractors for AMD classification in OCT using convolutional neural networks. InSignal Processing Conference (EUSIPCO), 2017 25th European 2017 Aug 28 (pp. 51-55). IEEE.

2. Ravenscroft D, Deng J, Xie X, Terry L, Margrain TH, North RV, Wood A. AMD Classification in Choroidal OCT Using Hierarchical Texton Mining. InInternational Conference on Advanced Concepts for Intelligent Vision Systems 2017 Sep 18 (pp. 237-248). Springer, Cham.

Chapter 2

Background

2.1 Deep Learning

Deep learning is currently at the forefront of computer vision with a wide range of applications. In this section we look at some of the standard techniques of deep learning before looking at some of the state-of-the-art architectures in this field.

2.1.1 Fully Connected Neural Networks

Neural networks (NN) are born out of an attempt to imitate the biological process of image recognition in the human brain in which a huge series of neurons pass electrical signals to the visual cortex of the brain. There are billions of neurons each connected to thousands of others. Impulses from input neurons arrive simultaneously at synapses which sum them to calculate the resultant nerve impulse to send onwards. Neural networks model this by using a set of weighted, connected layers arranged in a feed-forward network in which the weightings can be automatically optimised.

These networks are built on the Perceptron model [4], shown in Fig. 2.1. Each node receives a number of weighted inputs which are summed and passed to an activation function which calculates the value of the output. Sigmoid functions are commonly used which produce an output between 0 and 1.

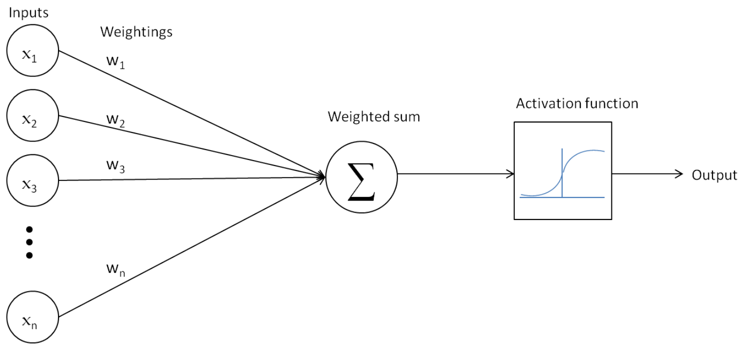


Figure 2.1: The Perceptron model.

These Perceptrons can be combined in a feed-forward network with outputs of all the nodes of one layer feeding into each node in the next layer creating a fully connected network. Any number of these layers can be added but parameters quickly increase as the network gets deeper. The output of the network can be a solitary node which produces a binary classification or a number of nodes equal to the number of classes.

The most important aspect of this model is that weightings can automatically be optimised using a technique called back-propagation. During training, inputs are fed forward through the network with their values at each stage changing according to weightings. At the end of the network is a layer which calculates error, commonly a Softmax layer is used. It compares the calculated output to the actual class and calculates the error for the input. Using the chain rule for differentials and an optimisation technique such as stochastic gradient descent it is possible to propagate the error back through the network and update all weightings. This allows the network to automatically calculate optimum weightings and effectively learn pertinent features from the input data.

2.1.2 Convolutional Neural Networks

Drawbacks of fully connected NNs is that features are not localised and the number of parameters rapidly increases as the network gets deeper, making them particularly unsuited to dealing with images.

Convolutional neural networks (CNN) are a variety of neural network which are suited to image recognition problems by overcoming the problems presented by the fully connected NNs [5]. The key innovation is the introduction of convolutional layers which are able to learn localised features and require far fewer parameters than fully connected NNs. These layers consist of a bank of small, locally receptive filters which are convolved across the whole input image. Each filter has a single set of weights rather than there being a weight for each node. This greatly reduces the number of parameters allowing much deeper networks to be created. The filters are also able to identify local features more easily by examining the relationship between pixels in smaller areas of an image rather than the whole image. Pooling layers are also common in convolutional networks which decrease spatial information by taking the average (or maximum or minimum) value over a local region of each feature map, again reducing computational expense. Additional layers such as regularisation layers and dropout layers are also common which decrease overfitting. Fully connected layers still often feature in convolutional networks, often appearing at the end of the network to feed into the classifier layer when there are fewer parameters and so are less computationally expensive. Back-propagation of error using stochastic gradient descent works in the same way as it does for fully connected NNs allowing the same automatic learning of features and classification.

2.1.3 State-of-the-art Deep Learning Architectures

In this subsection we will look at how the state-of-the-art deep learning algorithms have evolved over the last few years. ILSVRC (ImageNet Large-Sclale Visual

Recognition Challenge) [6] is an object recognition dataset which is used as a benchmark to measure the performance of deep learning algorithms. We will examine AlexNet, VGG, GoogleNet and ResNet which have, in turn, all produced the top results for this dataset.

AlexNet [7] marked the breakthrough of convolutional networks when it produced a top 5 error rate of 15.4% in ILSVRC 2012 which eclipsed all other entries, the next best being 26.2%. The proposed network consisted of 5 convolutional layers, max-pooling layers, dropout layers and 3 fully connected layers. The architecture consisted of two linked branches, as two GPUs were required due to the computationally expense, in a relatively simple feed-forward architecture.

VGG [8] presented a simple but deep convolutional architecture. It consisted of 13 convolutional layers each with filters of size 3×3 along with max-pooling layers and three fully connected layers at the end. It demonstrated the benefits of smaller filters which, when used in combination, have the same effect as larger filters. They also increase the number of filters in each convolutional layer following a pooling layer emphasising the idea of depth over large spatial dimensions.

GoogleNet [9] presented an architecture with greatly increased complexity but producing a top 5 error rate of just 6.7%. They improve utilisation of computing resources allowing the deeper and wider architecture. The main basis of the network is the use of nine Inception modules. As shown in Fig. 2.2 each module contains 1×1, 3×3 and 5×5 convolutional layers allowing the capture of dense and more spatially spread information. They also use 1×1 convolutional layers for dimensionality reduction to decrease computationally expense. Additional classifiers are attached to intermediate layers during training to strengthen gradient descent in such a deep network; these are removed during testing. This method also allows them to forego fully connected layers, which are usually the most computationally expensive part of the network.

ResNet is one of the state-of-the-art architectures used in deep learning which uses residual convolutional layers; that is, some outputs are reused, being com-

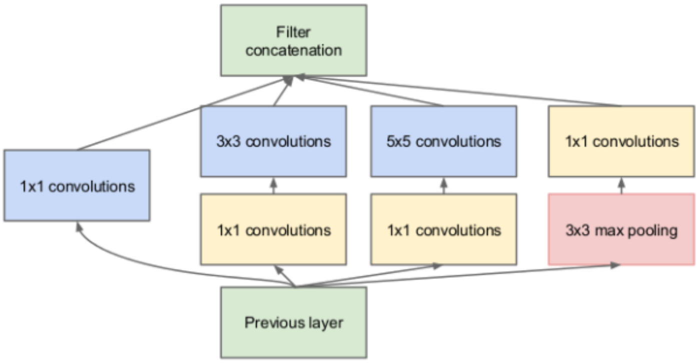


Figure 2.2: An Inception module using 1 × 1 convolutions for dimension reduction [9].

bined with outputs of later layers [10]. In ResNet, fewer fully connected layers are needed and, as these are the most computationally expensive layers, this allows much deeper networks to be built producing more accurate results. The fewer fully connected layers also results in less overfitting. This method can consist of architectures of varying numbers of layers and filter sizes depending on input size. The success of this method was first demonstrated on the ImageNet dataset [11] and is now widely used in many other aspects of visual computing including semantic segmentation [12], object detection [13] and contour detection [14]. The architecture consists of a series of residual blocks, as shown in Fig. 2.3. In each of these blocks the input x goes through a sequence of layers consisting of a convolutional layer, a ReLU layer and a second convolutional layer. The output of the second convolutional layer is then added to the original input. The residual block is essentially computing a term to add the original input which causes it a slight change. This is in contrast to traditional convolutional layers where each layer is a completely new representation which does not retain information about the input. At the start there is pooling layers to decrease spatial information before a series of residual blocks, one final average pooling layer and a fully convolutional layer which feeds into the Softmax layer for error calculation. In [10] a 152 layer network is used; this produced a 3.6% error rate for ILSVRC 2015 which comfortably surpassed the previous best set by GoogleNet of 6.7%.

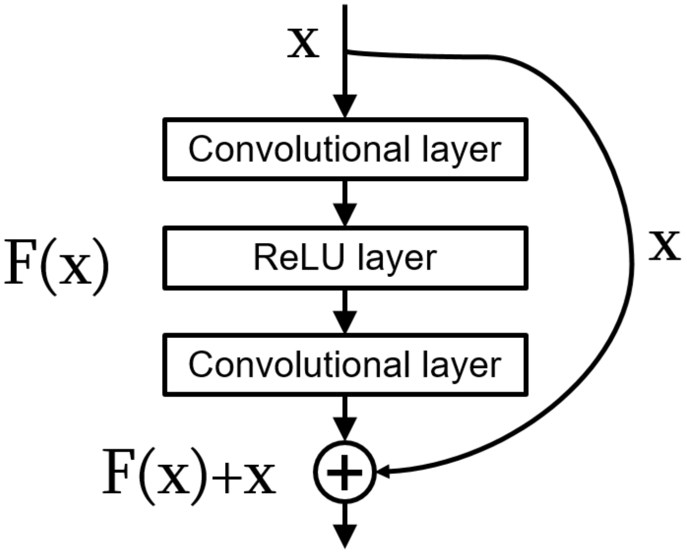


Figure 2.3: A residual block.

2.2 Traditional Texture Analysis

Texture analysis is an important problem in computer vision with applications in surface defection discovery [15] and image-based medical diagnosis [16]. It involves extracting features from an image, based on its textural appearance, which can then be used for classification. Hand-crafted features are designed for the specific problem which highlight the discriminative pattern for visual recognition task. Here we consider some of the leading techniques.

Local Binary Patterns (LBP) are a type of statistical method that works by comparing each pixel to its neighbours [17]. It takes a pixel as a centre and uses its grey level as a threshold for it to class its neighbours as 0 or 1. The centre pixel is then given a value which is a weighted sum of its binary neighbours:

$$\mathcal{L}_{P,R} = \sum_{p=0}^{P-1} sign(g_p - g_c)2^p \tag{2.1}$$

where g_c and g_p are the grey levels of the centre and neighbouring pixels, P is total number of neighbourhood pixels, R is radius and $sign$ is the binary value from thresholding. It is useful for texture classification as it is invariant to changes in illumination and rotation, and requires little computation.

Markov Random Field (MRF) models create an image by describing the relationship of an individual pixel's grey value with that of the grey values in its neighbourhood [18]. This is achieved by using a conditional probability distribution to describe local neighbourhood interactions.

A Gabor filter is a linear filter consisting of a Gaussian kernel function multiplied by a sinusoidal wave [19]. It has similarities between its feature descriptors and the stimulation mechanism of human visual system. They consist of two parts: an imaginary part and a real part. As they can extract features at different orientations and scales, they are a widely used technique.They have been successfully used in medical imaging [20, 21] outperforming other methods by extracting impulse responses from different scales and orientations.

All of these techniques are hand-crafted and have to be selected individually to fit the chosen dataset. Deep learning offers the opportunity to create self-learning algorithms which are generalisable and can be used on a range of different datasets.

2.3 Texture Analysis using Deep Learning

In Section 2.1 we explored deep learning techniques and demonstrated their success in a range of computer vision problems. Here we examine some of the applications of deep learning within texture analysis.

2.3.1 Texture Classification

Texture classification involves, given an image of a certain texture, being able to correctly categorise what it is. Being able to identify the pertinent features of a texture is important and the feature-learning ability of convolutional networks is well suited to this task.

Tivive [22] is one of the first to use convolutional networks in texture analysis. They propose using the CNN in an end-to-end framework in which it learns the features and provides a classification. They use a relatively shallow network with just two convolutional layers but are able to produce classification accuracies, on their 10 texture class dataset, similar to those achieved by, at the time, state-of-the-art techniques such as wavelets and Gabor filters.

Cimpoi [23] demonstrates that deep learning methods, including Improved Fisher Vectors and Deep Convolutional Activation Features (DeCAF), can be adapted from use in object recognition to texture recognition. The DeCAF features are obtained by using a pretrained CNN [24]. The remove the Softmax layer and fully connected layers with the adapted network producing a 4096 dimensional descriptor vector which is passed to a support vector machine for classification. Whilst these were generalisable models using networks pretrained for object recognition, they outperformed specialised texture descriptors. This demonstrates the capability of deep learning over traditional texture classification techniques.

Andrearczyk and Whelan [25] propose a convolutional network called Texture CNN. It has a simple architecture which exploits the ability of filter banks to effectively extract texture features. The network consists of a couple of convolution layers and a pooling layer with an energy layer added before the fully connected layers. It is smaller than commonly used convolutional networks but the energy layer assists in extracting texture information. It is tested across a range of different datasets demonstrating comparable or superior performance to

AlexNet [7] for texture recognition. It does not, however, compare its results to traditional texture analysis techniques. In [26] they show promising results using the Texture CNN in a biomedical imaging example. It presents the CNN proposed in [25] applied to datasets of liver tissue images and presents preliminary results which show an improvement on current state-of-the-art hand-crafted methods. They use images of 1388×1040 split and resized to 227×227 pixels. Whilst the results are limited it demonstrates the applicability of deep learning to medical image classification.

2.3.2 Texture Segmentation

Texture segmentation involves being able to partition images of mixed textures into the corresponding individual textures.

Cimpoi [27] introduces a new texture descriptor, FV-CNN, which uses Fisher Vector pooling of CNNs based on the texture descriptors of [23]. In this method filter banks are extracted from the convolutional network and Fisher Vector pooling performed on these. This approach is tested on the Flickr material dataset and MIT indoor scenes and produces state-of-the-art accuracy for texture, material and scene recognition. The CNN used is AlexNet as presented in [7] and pretrained on the ImageNet dataset. From this CNN the final convolutional layer is used with the Fisher Vector pooling local features densely removing global spatial information improving its ability to describe local texture.

In [28] a new approach of using convolutional nets for texture segmentation is explored. Several methods are employed to train convolutional networks to recognise and segment textures in various applications.They use networks which are fully convolutional based on the networks for semantic segmentation proposed in [29]. It consists of four convolutional layers with pooling layers connected using skip connections. The outputs of convolutional layers are fed into deeper layers of the network as well as the sequential layer. They test their architecture on on the Prague unsupervised texture segmentation dataset (ICPR contest 2012) in which they improve on the state-of-the-art results.

2.3.3 Texture Synthesis

Texture synthesis is one of the more researched aspects of texture analysis in respect to deep learning. Texture synthesis involves being able to develop images which have the appearance of a chosen texture. The filters of convolutional layers in a convolutional network can be used to identify important textural features and synthesise new images.

A new model of natural textures based on the feature spaces of CNNs optimised for object recognition is introduced in [30]. They use VGG-19 convolutional network [8] which uses small filters to find very localised features and is a state-of-the-art architecture for object recognition. Features of different sizes are extracted from the different layers of the trained convolutional network. The correlations between responses of these different layers are used as a spatial summary statistic. New images are generated by using gradient descent on a random image to produce the same stationary description derived from the spatial summary statistic. This is similar to [31] but using CNNs for the automatic generation of features and spatial descriptors. They demonstrate how their method outperforms [31] showing the potential benefits of self-learnt features.

[32] demonstrates that the filters from shallow CNNs can be effectively used as a model for natural textures. It shows that shallower networks can produce texture syntheses with a perceptual quality comparable to the state-of-the-art methods which require deep, multi-layered convolutional networks. They compare their results to [30], which uses a 19-layer VGG network, presenting visually similar results using a multi-scale single layer network, although they comment that they show less variability.

The work of Gatys [30] is expanded on in [33] by the exploration of the temporal dimension in texture synthesis. They substitute a 3D CNN for the 2D CNN used in the framework of Gatys. Using pretrained 3D ConvNets [34] they are able to compute correlation statistics on feature responses in the synthesis procedure with the added temporal dimension. Results presented demonstrate realistic dynamic textures can be synthesised.

In [35] they examine leveraging the knowledge learnt from large datasets and utilising it in smaller datasets. By using transfer learning, passing on learnt network weightings to be used as the initial setup, they were able to achieve results surpassing hand-crafted methods on datasets with little training data. It demonstrates the ability of CNNs to be able to learn the most germane features for texture analysis and their capacity for generalisation.

2.4 Summary

We introduced deep learning techniques looking at the neural networks and convolutional networks. We examined state-of-the-art architectures and explored why they have been successful and their applications.

The background demonstrates that using deep learning for texture recognition and classification is a effective technique and is able to outperform more traditional methods involving extracting hand-crafted features. Positive results are seen in texture synthesis, segmentation and, as we will explore, classification providing a sound basis for our research.

We note in particular a couple of techniques which have been touched upon which we look to examine in this work. Firstly, in [23] we see the use of a convolutional network to produce the feature descriptors which are then fed into a separate deep learning technique for classification. Also noteworthy is the observation that a majority of techniques using convolutional networks require shallow architectures to produce accurate results. These findings are used in the formulation of the methodologies we use in this thesis.

However, these techniques are widely used on well-examined textures which are clearly distinguishable to the human eye. The materials are generally well defined and distinct from each other whereas we will be exploring subtle nuances between textures with appearances which, to the human eye, are similar. Furthermore, the datasets have larger datasets with sufficiently large subsets of each texture to be learnt. We are dealing with a limited size of data with only 25 patients in each class. Additionally, it is not known if texture will be consistent, within each class, across the whole of the choroidal region or whether changes of texture in diseased eyes occur only in localised areas. Consequently, our task is more complicated than most current studies in deep learning for texture recognition.

Chapter 3

Dataset

3.1 Age-related Macular Degeneration

AMD is a progressive eye disease which is the leading cause of vision loss in the developed world. It is particularly prevalent amongst the elderly with 35% of over 80s in USA suffering from it [1]. The macula is a small area in the retina (see Fig. 3.3) which contains many photoreceptors and is responsible for central vision. This area is affected by AMD which consequently has an adverse effect on vision. It is a progressive disease; vision loss is minimal at an early stage but it can develop to one of two end stages: dry (geographic atrophy) or wet (neovascular AMD) [36, 37]. In dry, there is a gradual degradation of the retina due to the accumulation of fatty deposits. This occurs over a number of years and results in gradual vision loss. In wet, blood vessels break through the retinal pigment epithelium from the choroid into the retina. This leads to blood, fluids and lipids leaking into the retina. This results in blurry patches and vision distortion and can lead to complete loss of central vision in the affected eye in a matter of days. Early and accurate diagnosis with effective treatment can prevent it developing into an irreversible AMD stage and minimise damage to the retina layer and choroidal region.

Fig. 3.1 shows examples of the choroid Optical Coherence Tomography (OCT) images in different AMD categories. It is our hypothesis that the pathological progression of AMD has an effect on the shape and texture of the choroidal region due to changes in the choroidal vascular structure and this information is embedded in the OCT images. However, this hypothesis has not been fully studied due to the fact that the images obtained of the choroidal regions are very noisy and exhibit large variations from patient to patient. Acquiring a large cohort of patient data with labeled choroidal region is also a challenge, see examples in Fig. 3.1. As the first step towards automated diagnosis, in this work we study the feasibility of applying texture analysis to AMD classification using only choroidal regions.

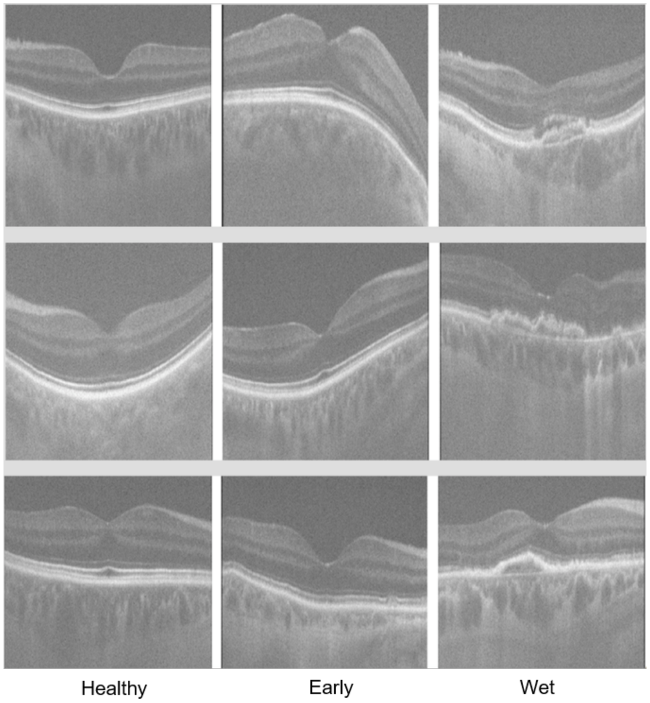


Figure 3.1: Examples of choroidal OCT scans for healthy, early AMD and wet AMD.

3.2 Optical Coherence Tomography

Optical Coherence Tomography (OCT) is currently the state-of-the-art technique used in opthalmology for imaging the deep regions of a patient's eye. It is less invasive than other techniques and produces more accurate images which has contributed to its recent increase in popularity.

Two of the previously favoured techniques were angiography and ultrasonography [38]. Angiography involves an injection of a coloured dye into veins in the patients' eyes with a series of photographs used to map the flow of blood in the eye. This process is, however, very invasive and does not allow accurate imaging of deeper regions of the eye as dye may causing staining as it advances through the blood vessels. A method of ultrasound can be used in which sound waves are bounced off the eye with the strength of response being used to calculate the depth of features. This is less intrusive than angiography but can be too low resolution to develop accurate maps, especially of deep structures such as the choroid.

OCT, as used in this study, produces the most accurate images of deep regions of the eye. The method involves using near-infrared wavelength light to produce accurate photographs [39, 40]. Using a longer wavelength than in traditional techniques allows a greater penetration of the eye allowing imaging to penetrate the retinal pigment epithelium and visualise the choroidal layer. The images obtained using this technique are sufficiently high resolution to allow the texture of this region to be analysed. As OCT is a reasonably new technology to be used for this purpose accurate images of the choroidal regions, until recently, have been meager. Resultantly, little research has been done into appearance and textural changes of the choroidal region in stages of AMD and whether it can be used for classification of these is largely untested.

3.3 Related Studies of Choroidal Diseases

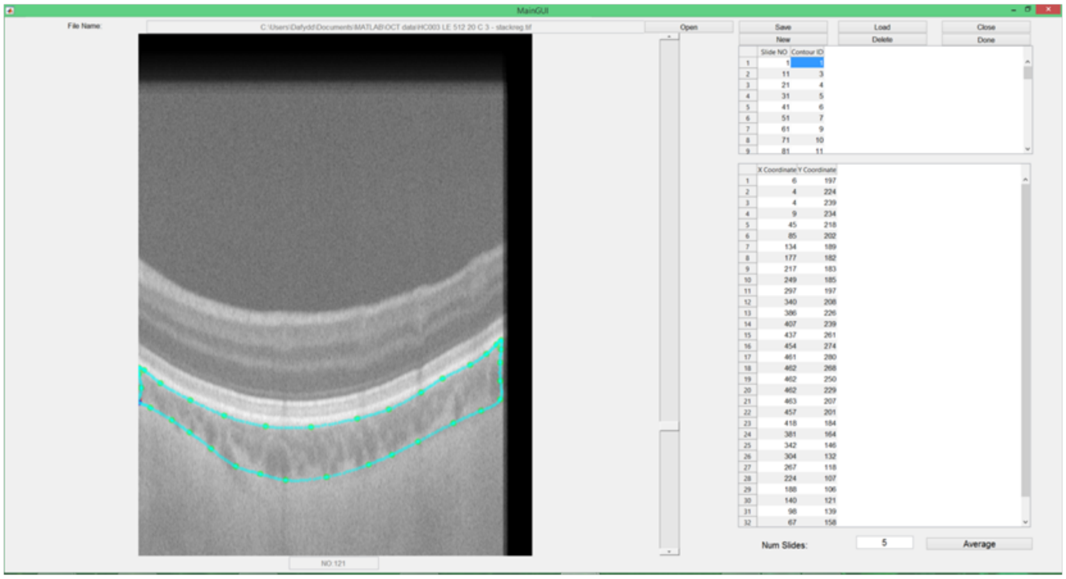
Priya et al. [41] proposed a machine learning approach for classifying AMD using colour retinal photographs, where hand-crafted features were extracted, such as retinal vessel density and average retinal vessel thickness. They claim to produce accuracies of 96% on their 100 image test dataset. However, they not not detail how their dataset was compiled and are vague on the exact methodology used. This method also involved manually calculating a number of features based on the preprocessed input image whereas our method will be fully automated. In [42] the abnormality measurements of Retinal Pigment Epithelium (RPE) layer, bubbles in Retinal Nerve Fiber Layer (RNFL) complex region and outer RNFL region near RPE layer were used to construct a binary discriminative model that classifies the images into AMD and Diabetic Macular Edema (DME). Farsiu et al. [43] used the thickness measurement of RPE Drusen Complex (RPEDC) and Total Retina (TR) as features to build a generalised linear model for AMD classification. They produced an area under the curve (AUC) of the receiving operating characterisitc (ROC) of 0.99. This differs from our work as it uses a larger section of the eye and requires semi-automatically calculating a number of features. Koprowski et al. [44] proposed a random forests based method to classify choroidal OCT images into predefined clinical conditions by extracting high level features, such as number of detected objects and average position of the centre of gravity, from low level texture information. This resulted in accuracies of 73%, 83% and 69% for the three disease classification classes.

These high level features heavily rely on high quality detection and segmentation results of blood vessel and other anatomical structures, which normally requires extra human resource. Designing hand-crafted filters is a time consuming and challenging task. More often than not, such techniques do not adapt well with data and also can not readily be implemented when input images are of inconsistent size and shape. There are three major difficulties of applying traditional hand-crafted filters to AMD classification problems using choroidal OCT images. Firstly, variations of local textural appearance within the choroid are very subtle and nearly random in high frequency bands, while such variations change across slices in low frequency bands, i.e. designing feature extractors that are able to capture the representative patterns is a non-trivial task. Secondly, pathological effects of AMD are not homogeneous in the choroidal regions. Textural features are thus highly non-uniform. Thirdly, the choroidal sections are irregular in size and shape across different subjects resulting in feature descriptors of arbitrary length. Based on these limitations we believe that using deep learning techniques for feature representation and classification can overcome these challenges.

3.4 Dataset

The dataset has been developed in collaboration with Cardiff University's Optometry department. It consists of 25 healthy eye scans from the control group, and 50 scans from AMD patients classified into one of two categories: early AMD and wet AMD. Therefore, for each of the three categories the dataset contains 25 eye scans preventing bias by ensuring there is no dominant class during training. In order to obtain high quality images, the long-wavelength (1040nm) OCT imaging technique is used to provide sufficient light penetration into the choroid structure. For each eye, a volume of 512×1024×512 pixels is produced by a 20°×20° volume scan. Each eye has its axial eye length (AEL) measured, and the images were scaled accordingly; this was done to control for errors in image scaling [45].

All samples were collected by the same operator and classified by three experienced optometrists into the pathological categories. Classifications were made by examining the shape and appearance of the retina based on an adapted version of an accepted and widely used clinical classification system [46]. We take these classifications to be the ground truth. In preprocessing, for each eye, the outline of the choroidal region was manually labelled on every tenth slice, hence meaning the dataset consisted of over 3,800 labelled slices. Automatic image segmentation has been shown to work in medical examples [47, 48, 49] but we chose manual segmentation to ensure accuracy and consistency. This process involves manually marking the outline of the choroid, as shown in the diagram. Shown, in Figure 3.4, is a snapshot of the toolkit used for the labelling. It shows the coordinates of each of the marked boundary points. The choroidal outline was easily identifiable meaning manual segmentation was reasonable. The labelling was shared between myself and Cardiff University's Optometry department. Examples were initially demonstrated by the team from Cardiff with the segmentations produced by myself also being validated by them.



File Name: C:\Users\Dafydd\Documents\MATLAB\OCT data\HC003 LE 512 20 C 3 - stackreg.tif

	Slide NO	Contour ID
1	1	1
2	11	3
3	21	4
4	31	5
5	41	6
6	51	7
7	61	9
8	71	10
9	81	11

	X Coordinate	Y Coordinate
1	6	197
2	4	224
3	4	239
4	9	234
5	45	218
6	85	202
7	134	189
8	177	182
9	217	183
10	249	185
11	297	197
12	340	208
13	386	226
14	407	239
15	437	261
16	454	274
17	461	280
18	462	268
19	462	250
20	462	229
21	463	207
22	457	201
23	418	184
24	381	164
25	342	146
26	304	132
27	267	118
28	224	107
29	188	106
30	140	121
31	98	139
32	67	158

Num Slides: 5

NO 121

Figure 3.2: GUI for choroid labelling

Fig. 3.3 shows examples of labelled OCT scans of the three categories. From each image the closed curve created by the labels was extracted leaving just the choroidal layer for each slice.

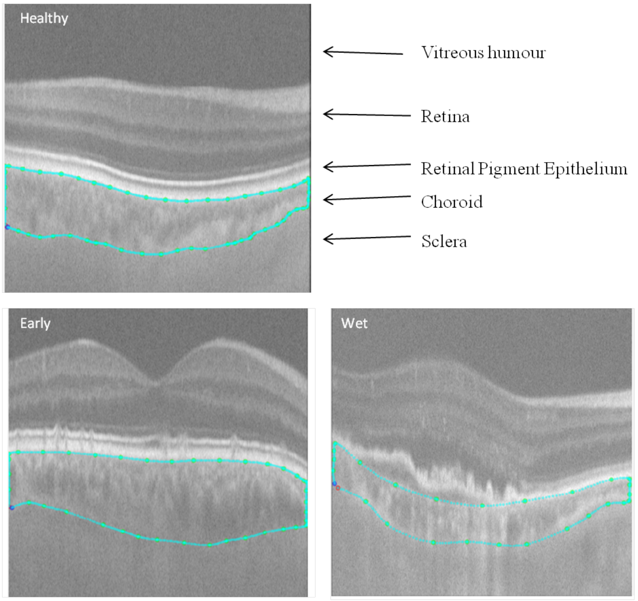


Figure 3.3: Examples of labelled OCT scans for each of the three classes with visible signs of pathology within the retina. The outline of the choroid is shown in each image with various parts of the eye labelled in the Healthy image.

3.5 Summary

We have examined the widespread nature of AMD and its adverse effects especially extensive amongst the elderly. The ability to treat it effectively given timely diagnosis highlights the importance of efficient and accurate diagnosis. The disease affects the choroidal layer of the eye; an area for which recently, due to technological advances, accurate images have become able to be easily obtained. Whilst the field of AMD is generally well researched, exploration into the relationship between the disease and choroidal texture is limited.

The dataset we are to use has been developed using state-of-the-art technology to produce images of the deep sections of the patients' eyes with manual labelling used to extract the choroidal layer specifically. This produces a dataset of high quality images. 75 patients are each grouped into one of three evenly sized groups: healthy, early AMD and wet AMD. By using every 50 cross-sectional slices for each eye this produces 3,800 labelled slices in total. However, size could be a restrictive factor on performance with only 25 patients per group potentially limiting the intra-group variety. Additionally, by extracting only the choroidal layer each of the images will be differently shaped which presents another challenge.

Chapter 4

Hand-crafted Features

The concentration of this thesis is to explore the performance of deep learning techniques. However, it is therefore important to be able to compare the results from these methods to those obtained from traditional methodologies. In this section we use hand-crafted methods for feature extraction with these then being classified using a fully connected neural network or random forests.

4.1 Gabor Filters

The use of Gabor filters for feature extraction has been successfully used in texture recognition problems [19, 50, 51] and in medical image classification [20, 21, 52]. They perform strongly on texture analysis problems due to the ability to extract features from different scales and directionality. These are key features of random texture and being able to model these characteristics is the reason they produce strong results and are so widely used. A filter is effectively a matrix which is convolved across the image being applied to each pixel and its neighbours. This has the effect of smoothing the image and making edges clearer. Based on its success in other texture problems in medical imaging we used Gabor filters as a base against which to compare our deep learning methods.

4.2 Methodology

Our method of feature extraction is to use a method similar to that proposed by Jain et al. [53]. We use a 2D Gabor filter; this function consists of a sinusoidal wave of some frequency and orientation multiplied by a Gaussian function. This Gabor filter is given by:

$$h(x, y) = exp\left\{ -\frac{1}{2}\left[\frac{x^2}{\sigma_x^2} + \frac{y^2}{\sigma_y^2}\right]\right\} cos(2\pi u_0 x + \phi)$$

where u_0 represents the frequency and ϕ the phase of the sinusoidal wave, and σ_x and σ_y represent space constants along the x and y axes respectively.

The set of filters is convolved across the images individually to obtain a set of filtered images. The filter is applied to each pixel in turn, therefore, producing an output of the same dimensions as the input image. A response for each image for each filter is produced. The filter responses are then passed to a histogram to decrease the spatial information and to improve generalisation. The histograms of each classifier are combined to produce a single feature vector for each image. The Gabor filter banks used 2D Gabor filters of 39x39 pixels with 40 filters consisting of 5 different sizes and 8 orientations. From each of these filters, features were calculated and grouped into 11 bins, with histograms combined across the different filters. This produces a feature vector for each image with 440 values. These were then passed on to classifiers, random forests and neural networks, which were each applied independently in a supervised manner to produce classification accuracies.

4.3 Results

The results of this method for 10-fold and 2-fold cross validation are presented in Tables 4.1 and 4.2 respectively. We tested using both simple fully connected neural networks and random forests as classifiers. The random forest consisted of

50 random decision trees, and the neural networks contained two hidden layers with 200 and 40 nodes respectively. For each method of validation the training and testing process was iterated 10 times with the demonstrated results the combination of these.

The results demonstrate that a reasonable discrimination can be made between the three groups with top accuracies of 76.4% and 74.3% being found for 10-fold and 2-fold respectively, which is significantly better than random chance of 33.3%. As expected 10-fold results are better than 2-fold due to the higher ratio between the sizes of the training and testing sets. This provides a base level for feature extraction using hand-crafted methods. This can then be used to provide a quantitative comparison between this approach and the deep learning approaches we propose.

		Healthy	**Early AMD**	**Wet AMD**	**Avg.**
NN	**Healthy**	**80.5**	9.6	12.3	**75.7**
	Early AMD	6.8	**75.6**	16.8	
	Wet AMD	12.7	14.8	**70.9**	
RFC	**Healthy**	**84.1**	11.5	18.7	**76.4**
	Early AMD	6.6	**77.9**	14.2	
	Wet AMD	9.3	10.6	**67.1**	

Table 4.1: Confusion matrices of classifiers using Gabor filters for 10-fold cross validation (%)

		Healthy	**Early AMD**	**Wet AMD**	**Avg.**
NN	**Healthy**	**74.6**	6.5	14.7	**72.0**
	Early AMD	8.6	**74.2**	18.1	
	Wet AMD	16.8	19.2	**67.2**	
RFC	**Healthy**	**80.8**	9.9	18.6	**74.3**
	Early AMD	7.5	**76.5**	15.9	
	Wet AMD	11.7	13.5	**65.5**	

Table 4.2: Confusion matrices of classifiers using Gabor filters for 2-fold cross validation (%)

Chapter 5

Patch-based Approach

One of the challenges presented by our dataset is the irregularity in size and shape of the choroidal images, not only from patient to patient, but also between each of the slices of an individual patient's eye. Due to the design of convolutional networks it is necessary that inputs are all of the same size and shape; this ensures the back-propagation algorithm is able to work correctly. To overcome this problem we propose extracting a set of equally sized patches from the choroidal layer of every OCT scan which can then be used with convolutional networks.

In this chapter we firstly examine the possibility of using a convolutional network for feature extraction. A bank of filters, learnt in a convolutional network, are used to produce feature descriptors from the image dataset, in Section 5.1. We then investigate methods to generalise these feature descriptors and learn their spatial distributions, in Sections 5.2 and 5.3, before passing on to machine learning techniques for classification.

5.1 Feature Extraction

CNNs combine both feature representation learning and supervised discrimination into a uniform end-to-end training framework, which have become very popular in recent years and produce the top results for many machine vision problems [7, 8, 9]. In this method, a CNN is introduced to hierarchically learn the textural features in a supervised manner. In convolutional layers, a bank of locally receptive filters convolve across the input image to form visual evidences for prediction layers at the forward pass stage. At the backward pass stage, these filters are automatically optimised via back-propagating the prediction error of the forward pass. Fully connected layers are also included in the network; in these all nodes from one layer are connected to all nodes in the next with weightings updated in the same way. This allows pertinent localised features to be more easily identified. Fig. 5.1 and Table 5.1 show the architecture details of the proposed CNN. Due to the irregular shape of the choroidal region (see Fig. 3.1), it is difficult to extract large local patches without including other structures. As such, local patches of consistent dimension are extracted from the slices to train the network. Ten patches of 48×48 pixels are extracted randomly from each annotated slice, with overlap. Each patch is given the same classification as the slice to which it belongs. In order to interpret the low level textural features learnt through the discrimination task, only one convolutional layer is used. This is consistent with other works which have found success in using shallow convolutional networks for texture analysis. Networks that are used for natural image recognition tasks generally have small kernel sizes, such as 3×3, as natural images have much sharper corners and higher contrast compared to medical imaging. In our case, 40 filter kernels with size of 9×9 are used in order to identify the discriminative patterns in low frequency bands. The first layer of the CNN consists, therefore, of 40 filters of size 9×9; these are initially randomised to very small values but the weightings are optimised through repeated back-propagation. The CNN is trained on the set of extracted patches and their associated classification. Figure 5.2 shows examples of the self-learnt filters kernels from this approach.

We tested using the convolutional network for end-to-end learning; producing a classification directly for each input. Tables 5.2 and 5.3 show the per-patch

No	Type	Parameter
0	Input	48×48×1 images scaled to [0,1]
1	Conv.	40 9×9 filters with stride 1
2	ReLU	Rectified linear unit
3	F.C.	Fully connected with 128 outputs
4	F.C.	Fully connected with 128 outputs
5	F.C.	Fully connected with 3 outputs
6	Softmax	Softmax probability for multi-classes

Table 5.1: The parameters of the proposed CNN architecture.

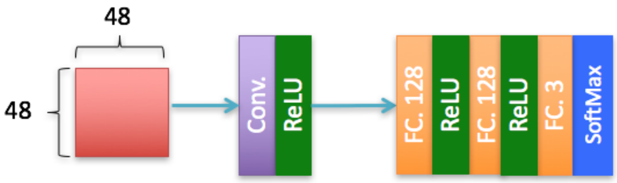


Figure 5.1: The network architecture of the proposed CNN.

accuracy achieved for 10-fold and 2-fold validation. These results demonstrate that using this setup directly on the patches is unable to extract sufficient information for accurate classification with classification biased towards the Healthy class. Dealing with such small patches it is unable to extract important spatial information about the choroid as a whole and has to rely on using low-level features.

In the next two sections we propose methods which use generalisation techniques, in combination with using the learnt CNN filters as feature extractors, to produce high-level features which should produce more accurate classification.

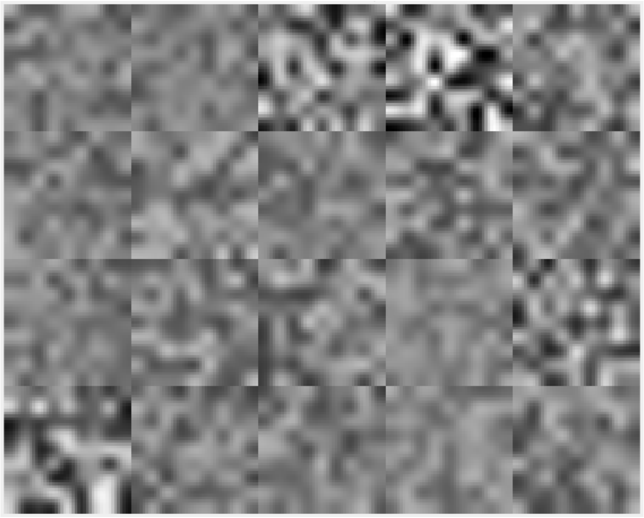


Figure 5.2: Examples of self-learnt filter kernels using the CNN.

	Healthy	**Early AMD**	**Wet AMD**
Healthy	**80.7**	**80.0**	**80.4**
Early AMD	19.2	20.0	19.4
Wet AMD	0.06	0	0.20

Table 5.2: Confusion matrix of 10-fold CNN classification (%)

	Healthy	**Early AMD**	**Wet AMD**
Healthy	**100**	**100**	**100**
Early AMD	0	0	0
Wet AMD	0	0	0

Table 5.3: Confusion matrix of 2-fold CNN classification (%)

5.2 Histogram Generalisation

CNNs are usually powerful discriminators but we introduce histograms for feature generalisation. The CNNs are only trained on patches extracted from the choroid; using a relatively small area presents a challenge for developing an accurate model. By introducing histograms, we can adapt the method to use the whole choroidal region rather than patches. Convolving the learnt filters across the entire slice and then taking a histogram of the choroidal region allows a description of the whole of the annotated region rather than a subsection of it.

5.2.1 Methodology

Fig. 5.2 shows examples of learnt filter kernels from the convolutional layer developed in Section 5.1. The bank of learnt filters are, in turn, convolved across the images of the extracted choroids. Convolving the filters across the images is a form of linear filtering which provides the responses of the kernel patterns. The annotated choroidal regions have variations in size and shape, therefore, outputs of the convolutional response vectors will be of inconsistent length. It is necessary to have the same sized feature descriptors for all slices in order to train the classifiers. To achieve this, the histogram of filter kernel responses of the annotated region is computed to be used as the feature descriptors rather than using local features directly. In addition, to allow images of different sizes and shapes to have the same length of feature output, histogram based descriptors produce a representation of the distribution of kernel responses which greatly improves the generalisation ability. Especially, in our case, the kernel filters are self-learnt through performing discrimination tasks, which generally is difficult to link to the pathological changes, and interpret their physical meanings (see Fig. 5.2). The histogram descriptor provides the quantified statistical measurements of the response distribution of given kernel patterns, which helps to identify the discriminative features. The histogram descriptor is calculated for each of the different filters with the results concatenated to produce one feature vector for each image. Fig. 5.3 shows examples of histogram descriptors of different AMD classes produced by the top 3 most discriminative filter kernels. This image demonstrates the differences between each class with the healthy and wet AMD classes being most distinct, whereas the early AMD and wet AMD are most similar.

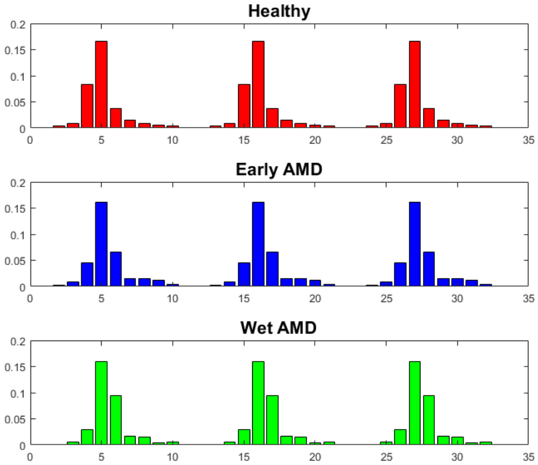


Figure 5.3: Examples of histogram descriptors of different AMD classes from top 3 most discriminative filter kernels.

5.2.2 Results

The convolution of each of the learnt filters was calculated for each image and grouped into 11 bins to produce the histogram. The histograms across the different filters are concatenated to produce a single feature vector for each image. As each filter bank consisted of 40 different filters this produces a feature vector for each image consisting of 440 values. Then, each of the classifiers were applied independently. To evaluate the discriminative power of proposed feature descriptors machine learning classifiers are employed to distinguish different AMD stages, such as: neural networks and random forests. In order to compare the discriminative power of histogram descriptors with the features of filter kernel responses, the traditional neural network is used, which consists of two fully connected layers. Random forest (RF) is an ensemble method which combines a number of weak classifiers to create an accurate predictive model [54, 55]. It averages the results of multiple decision trees, each of which consists of a set of recursive binary splits with leaf nodes assigning a probability of the training sample belonging to each class. The variable importance is evaluated during the training process through permutation, which ranks the discriminative power of learnt filter kernels. The random forest consisted of 50 random decision trees and the neural networks contained two hidden layers with 200 and 40 nodes respectively. For each method of validation the training and testing process was iterated 10 times with the demonstrated results the combination of these.

To perform AMD classification, 10-fold and 2-fold cross validations were used. For 10-fold cross validation the whole dataset was split into ten randomly sampled, evenly sized groups with an equal numbers of slices from each eye. One subset was held for testing whilst the other nine were used for training. This training set was used for learning the filters in the CNN and to train the classifiers. Table 5.4 shows the results of 10-fold classification for the three classifiers. Neural networks and random forests achieved correct classification accuracies of 83.3% and 66.2% respectively. Neural networks were decidedly the most accurate. They are useful for learning the hierarchical structure of features. Table 5.2 shows the result of using the CNN directly for feature learning and classification, where on average 33.6% is achieved which is significantly lower than the histogram descriptor (83.3% on average in Table 5.4). It strongly suggests that the histogram descriptor improves the discriminative power by a large margin. The primitive filter kernels shown in Fig. 5.2 is rather noisy and tends to appear random, however, in Fig. 5.3, the distributions of their responses are discriminative. Comparing these results to those obtained from using Gabor filters (Table 4.1) we see that this histogram method produced a higher accuracy, 83.3%, than the

Gabor method produced, 76.4%. This demonstrates the benefits of our machine learning approach compared to the hand-crafted approach for feature extraction.

The results of 2-fold cross validation of our proposed method are summarised in Table 5.5, with Table 5.3 showing the results through classifying using the CNN only. The CNN only method is unable to distinguish between stages of AMD whereas the respective prediction accuracies for the NNs and RFs were 76.9% and 54.3%. The accuracy was expected to decline across all classifiers due to the relative decrease in the size of the training set. However, a similar pattern occurs in which using the neural network as a classifier produces greater accuracy than the random forests. Furthermore, the highest accuracy of 76.9% is greater than the 74.3% obtained using Gabor filters for feature extraction (Table 4.2). The results suggest the feasibility of our approach for detecting textural changes in the choroid from which stages of AMD can be classified.

		Healthy	**Early AMD**	**Wet AMD**	**Avg.**
NN	**Healthy**	**81.2**	10.6	5.2	**83.3**
	Early AMD	11.1	**80.9**	7.0	
	Wet AMD	7.7	8.6	**87.8**	
RFC	**Healthy**	**59.8**	22.0	11.9	**66.2**
	Early AMD	24.3	**63.0**	12.4	
	Wet AMD	15.8	15.0	**75.7**	

Table 5.4: Confusion matrices of classifiers using histogram feature descriptors for 10-fold cross validation (%)

		Healthy	**Early AMD**	**Wet AMD**	**Avg.**
NN	**Healthy**	**73.9**	15.1	9.0	**76.9**
	Early AMD	12.7	**72.2**	6.3	
	Wet AMD	13.3	12.7	**84.6**	
RFC	**Healthy**	**49.6**	28.1	20.1	**54.5**
	Early AMD	30.3	**52.6**	18.6	
	Wet AMD	20.1	19.4	**61.4**	

Table 5.5: Confusion matrices of classifiers using histogram feature descriptors for 2-fold cross validation (%)

5.3 Texton Mining

The method suggested in Section 5.2 utilises a convolutional network to automatically learn a set of low level primitive filter kernels with the discriminative power generalised by using a histogram. In this chapter we build on this technique introducing clustering and Local Binary Patterns (LBP) to extract statistical and spatial information whose distribution is used to develop high level features suitable for classification.

Mining discriminative feature descriptors is the key component of designing an efficient visual recognition model for AMD stage classification using choroidal OCT images. The primitive low-level features are automatically learnt using a CNN, where the convolutional filter kernels are learnt via a supervised discriminative training procedure. Textons can then be inferred by clustering the image responses of learnt filter kernels, where the cluster centres form the texton dictionary. The spatial distribution of mined textons is extracted using LBPs.

Patch-based local textural features are then generalised to regional feature descriptors using histograms over the region of interest, which provide high-level features as a representation of the local distribution of the LBPs. Supervised classification is then carried out using machine learning techniques to classify input images into different AMD stages based on the texton feature descriptors that are mined hierarchically.

5.3.1 Methodology

Spatial Texton Descriptor

In this method, we introduce additional steps to explore both statistical and spatial distribution of the primitive texture feature that are produced from CNN. Firstly, as in Section 5.2, the bank of filters learnt in Section 5.1 are convolved across the images of the extracted choroids to produce a vector of filter responses for each image. The texture is modelled by the distribution of filter responses, these can be represented by textons (cluster centres) which can be used to create a texture model [56]. K-Means clustering is used to develop the set of textons, which can be used to label all filter responses with each observation assigned to the partition with the closest mean. The textons group the textural features into a compact representation via examining the statistical distribution of filter response, which removes the subtle variations at high frequency bands. These textons are more robust than the raw filter responses. However, for OCT retina image (see Fig. 3.1), the primitive texture appearances of choroidal region learnt from CNN are rather noisy and do not well form structural patterns (see Fig. 5.2). In order to overcome this difficulty, the spatial distribution of these mined textons is introduced which represents the patterns of local arrangement of mined texton. In the spatial domain, the local correlation of those textons can be further generalised in a hierarchical manner, which are more representative and informative for the classification task. In this method, LBPs are used to represent the spatial distribution of textons. Spatial features look for texture elements, known as texture primitives, which are extracted to create a representation that maps their regional locations. It looks for regular or repeated patterns of texture elements in the image, and learn spatial information by comparing each pixel to its neighbours and assigning each a binary value [57]. In this work, a texture unit is the central value in a 3×3 neighbourhood and is represented by the 8 elements that surround it. Each is assigned a binary value with the centre pixel acting as a threshold and are multiplied by predefined weightings based on the pixel location. The results of the eight neighbouring pixels are summed and this value is assigned to the texture unit. A value for each pixel is calculated meaning the response output has the same dimensions as the input.

Regional Texton Generalisation

As the CNNs are trained only on relatively small patches extracted from the choroidal regions, a higher level descriptor is required to make predictions on image level. We thus convolve the learnt CNN filters across the entire choroidal regions. Note that this is different to conventional texton learning, where kernel filters are pre-defined and static. The kernels in our method is data driven and dynamic. As the choroidal regions vary in size and shape, demonstrated in Fig. 3.1, which leads to varied length of LBP feature vectors. In order to train discriminative classifiers, it is desirable to obtain feature vectors of uniform length. Thus, a regional texton generalisation is carried out via computing the histogram of LBP feature vectors of the annotated region. The histogram based descriptors produce a representation of the distribution of responses which also improves the generalisation ability. For each of the filters a histogram is calculated with each LBP response being grouped into one of 59 bins depending on its value. The number of bins was calculated using the formula $P \times (P - 1) + 3$ where P is the number of neighbours, 8. The histogram descriptor is calculated for each of the different filters independently, and the results are concatenated to produce one feature vector for each image. Fig. 5.4 shows examples of histogram descriptors of different AMD classes produced by the top 3 most discriminative filter kernels. In Fig. 5.4, it is obvious that the differences between 3 AMD classes are distinct, although healthy and early AMD classes show some similarities, which is consistent with the clinical interpretation.

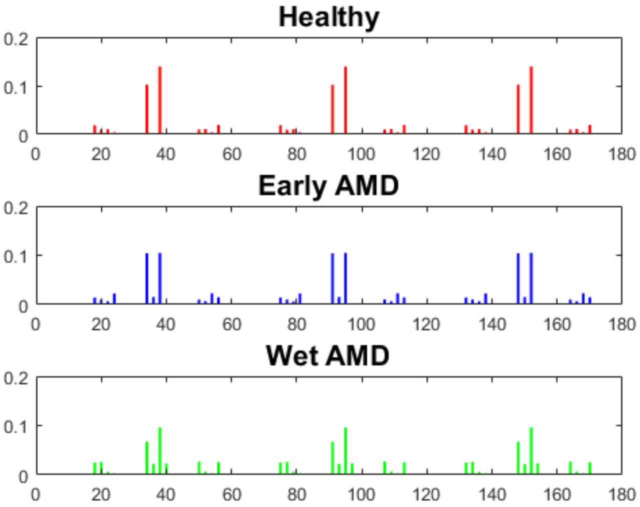


Figure 5.4: Examples of **texton** descriptors of different AMD classes from top 3 most discriminative filter kernels.

5.3.2 Results

K-means clustering was computed using 10 cluster centres. LBP used a neighbourhood of 8 pixels for value calculations. The number of bins for the histogram of LBP responses is calculated as $(P \times (P-1)+3)$, where P is the number of of neighbours, resulting in 59 bins. A histogram is calculated for each of the 40 filters with the results concatenated to produce a feature vector of 2360 values for each image. Then each of the classifiers were applied independently. The random forest consisted of 50 random decision trees, and the neural networks contained two hidden layers with 200 and 40 nodes respectively. For each method of validation the training and testing process was iterated 10 times with the demonstrated results the combination of these.

To perform classification, 10-fold and 2-fold cross validations were used. Tables 5.2 and 5.3 show the result of using the kernel feature learnt through CNN only, where on average 33.6% and 33.3% are achieved for 10-fold and 2-fold respectively, where the classification is dominated by the control group, and the prediction is nearly selected by random. Therefore, it strongly suggests that using the feature learnt from CNN only is unable to distinguish between different stages of AMD. Table 5.6 shows the results of 10-fold classification for three classifiers. NNs and RFs achieved correct classification accuracies of 78.5% and 87.8% respectively. There was a significant difference between the accuracy achieved using NNs as the classifier compared to RFs. NNs perform well when learning hierarchical structures of features directly from the raw input image. However, we develop the feature descriptors through learnt filters, spatial descriptors and histograms. As such, discriminative models which find a separation boundary between classes can be expected to outperform generalisation models. This represents an improvement in performance from our previous method (Section 5.2) which had a highest accuracy, for 10-fold validation, of 83.3%, as shown in Table 5.4. In this method, RFs outperform the NNs for classification whereas it was the reverse for the results from the previous method. This is as the features developed in this method are more high-level. By using clustering and LBPs we were able to extract spatial and statistical information about the images allowing discriminative classifiers to more easily differentiate between the classes.

The results of 2-fold cross validation of our proposed method are summarised in Table 5.7, where the respective prediction accuracies for the NNs and RFs were 75.0% and 85.2%. The accuracy was expected to decline across all classifiers due to the relative decrease in the size of the training set. However, a similar pattern occurs in which using the random forest as a classifier produces greater accuracy than the neural network. The hierarchical texton mining produces a more compact feature descriptor which enables a separable boundary to be found. Consistently with the 10-fold results, the highest accuracy of 85.2% surpasses the highest accuracy seen for the previous method, Table 5.5, of 76.9%.

In addition, the distinct accuracy differences between Tables 5.6 and 5.7, and Tables 5.2 and 5.3 show that the proposed feature descriptor improves the discriminative power by a large margin. From a feature selection perspective, the primitive filter kernels shown in Fig. 5.2 is rather noisy and tends to appear random, however, in Fig. 5.4, the distributions of their responses are far more discriminative. The results demonstrate a superior ability to develop high-level features over the previous method. This allows better classification results to be obtained as the boundaries separating the classes can be more accurately formed.

We compare our results to those obtained in other studies, as described in Section 3.3. Priya [41] claims to produce accuracies of 96% on their test set of colour retinal photographs when classifying AMD into one of three classifications: healthy, dry and wet. They do this by manually calculating features including retinal vessel density and average retinal vessel thickness. Whilst these results appear strong they provide no information about their dataset or how it was collected. Due to the lack of information about their data and their methodology it is difficult to say how reliable and reproducible these results would be. Koprowski [44] used random forests to classify choroidal OCT images into three different eye disease classification classes by extracting high level features, such as number of detected objects and average position of the centre of gravity. This resulted in accuracies of 73%, 83% and 69% for the three disease classification classes on the test set containing 20% of the dataset. This has similarities to our method in that it is classifying choroidal OCT images into one of three different categories, although the categories are different to ours. Our method, for 2-fold, produces a best accuracy of 85.2% which is better than all the classification accuracies in [44]. This suggests that our approach to learning the feature extractors can perform better than using manually calculated features, although as different datasets are used a categorical comparison is not possible. It is difficult to accurately draw comparisons to other works as this requires reasonable similarity between the goals and datasets. We, therefore, also compare our machine learning method to the handcrafted approach we performed on our dataset.

Comparing the 10-fold results (Table 5.6) to those obtained from using Gabor filters (Table 4.1) we see the superior performance of this texton mining methodology. Using NN and RF classifiers, this method produced higher accuracies, 78.5% and 87.8%, than similarly obtained using the Gabor filters as feature extractors, 75.7% and 76.4%. This demonstrates the benefits of machine learning methods for developing a set of useful features. By learning the feature extractors, rather than relying on hand-crafted ones, we are able to extract a more discriminative set of features. By selecting more pertinent features the ability to correctly classify unknown images increases. Furthermore, the 2-fold accuracies (Table 5.7) for NNs and RFs of 75.0% and 85.2% are greater than the 72.0% and 74.3% obtained using Gabor filters for feature extraction (Table 4.2). This is consistent with the 10-fold results showing the superior ability of machine learning feature extraction for this problem.

		Healthy	**Early AMD**	**Wet AMD**	**Avg.**
NN	**Healthy**	**74.3**	14.6	10.2	**78.5**
	Early AMD	16.3	**78.2**	6.8	
	Wet AMD	9.4	7.1	**83.0**	
RFC	**Healthy**	**84.2**	8.2	4.6	**87.8**
	Early AMD	9.9	**87.5**	3.5	
	Wet AMD	5.8	4.3	**91.8**	

Table 5.6: Confusion matrices of 10-fold cross validation **with the proposed feature descriptors** (%)

		Healthy	**Early AMD**	**Wet AMD**	**Avg.**
NN	**Healthy**	**65.4**	19.1	11.3	**75.0**
	Early AMD	22.5	**75.8**	4.8	
	Wet AMD	12.1	5.1	**83.9**	
RFC	**Healthy**	**81.9**	10.4	6.2	**85.2**
	Early AMD	10.7	**84.2**	4.4	
	Wet AMD	7.5	5.4	**89.4**	

Table 5.7: Confusion matrices of 2-fold cross validation **with the proposed feature descriptors** (%)

Chapter 6

End-to-end Histogram Framework

Using a patch-based approach allows us to be able to apply deep learning techniques to the irregularly shaped training set. We have shown in the previous chapter that patch-based methods show promising results in multi-class validation. However, in leave-one-eye-out validation, with a limited sized dataset, the challenge is more difficult. This is due to the limitations on the amount of information which can be extracted from a single patch. The patches, whilst taken randomly, must be fully contained by the region of interest and this could lead to systematic areas, particularly around edges, being excluded. This is exacerbated by the labelling process which allows only a single label to be applied to each eye while each individual slice may display varying degrees of AMD characteristics. This is even more exaggerated by extracting relatively small patches from the slice, many of which will not show any signs of disease when categorised into one of the diseased classes. We addressed this issue by introducing elements of generalisation after using the learnt features for feature extraction. These, however, are not optimisable whilst training the CNN. In this chapter we propose a novel method to include generalisation using histograms within a learnable end-to-end convolutional network. A similar technique is used in [58] to assist with semantic segmentation and object detection. They, however, use a different basis function and require inputs to be likelihood maps of the same size. Our method allows the whole irregular image to be utilised. The benefits of using a deeper network are also increased and we use state-of-the-art ResNet in conjunction with our histogram layers to produce improved results with leave-one-patient-out validation. We present a block of layers which act as a histogram using standard deep learning layers and a new masking layer to allow the irregularly shaped images to be used.

6.1 Histograms

Histograms are a form of generalisation which decrease the information in large volumes of data by splitting it into a predefined number of bins according to the value of each datum. This allows datasets of different sizes to be converted into descriptors of consistent size. This attribute will allow us to be able to aggregate the pixel data from each of the irregularly sized choroidal layers and produce a histogram of a consistent predefined size which describes it. A histogram can be defined using the formula:

$$f_b(x_i) = \begin{cases} 1, & \mu_b - \frac{w}{2} < x_i \leq \mu_b + \frac{w}{2}, \\ 0, & \text{otherwise} \end{cases} \tag{6.1}$$

Each input value, x_i, is categorised into one of B bins. For the b^{th} bin x_i is given a value of 1 if it falls in that bin's range, namely $[\mu_b - \frac{w}{2}, \mu_b + \frac{w}{2}]$, else it is assigned a value of 0. Each value is assigned to one, and no more than one, of the bins.

6.2 Differentiable Histogram

A requirement of the layers of a convolutional network is that they are differentiable; this allows stochastic gradient descent to be able to update weightings by back-propagating the error through the network during training. The above Equation 6.1 is not continuously differentiable and so cannot be used for end-to-end training within a convolutional network. To achieve this it is necessary to redefine the histogram into a differentiable equation. The b^{th} bin of the histogram can be modelled by a piecewise linear basis function $f_b(x)$:

$$f_b(x) = max\{0, w_b - \mid x - \mu_b \mid\} \tag{6.2}$$

where μ_b is the bin centre for the b^{th} bin, and w_b is the bin width. If the x value falls into the b^{th} bin, within the interval $[\mu_b - w_b, \mu_b + w_b]$, then this acts as a vote for this bin with weight $f_b(x)$. This allows each value of x to vote for more than one bin with the weight of the vote a measure of the distance from the bin centre. This produces a feature vector of b elements. Fig. 6.1 demonstrates an example in which $f_b(x)$ produces positive values of 0.08 and 0.12 for the bins centred at 0.6 and 0.8 and zero for the other bins.

A key necessity of neural networks to be able to automatically learn weightings is that the layers must be differentiable so that errors can be back-propagated through the network. Our proposed linear basis function, Equation 6.2, is piecewise differentiable. This allows the function to be broken into layers, each of which allow back-propagation. This enables the μ_b and w_b to be learnable during training. The bin centres and bin widths can be independently learnt using stochastic gradient descent meaning that a histogram contained within the convolutional network framework is itself learnable.

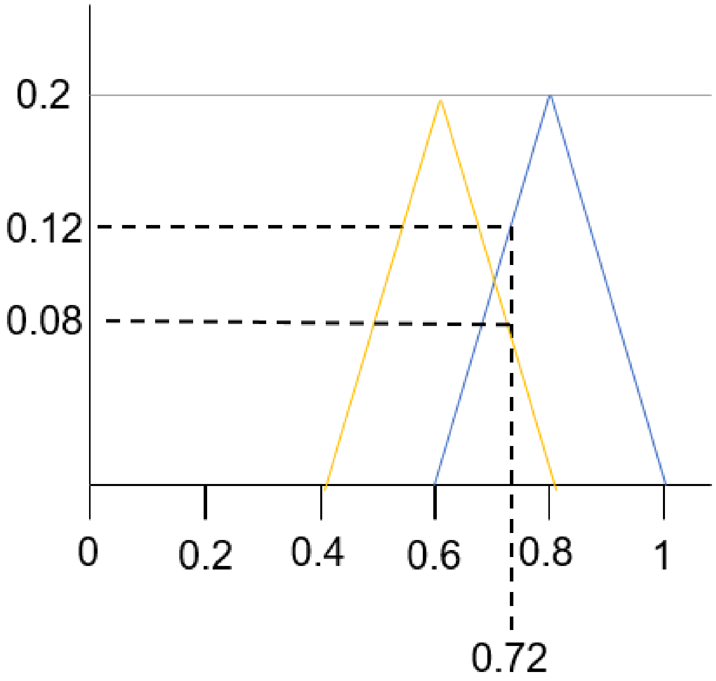


Figure 6.1: An example of the histogram linear basis for a histogram with 6 bins. Given a value of $x_i = 0.72$ it gives votes for bins 4 and 5 with respective weightings of 0.08 and 0.12.

6.3 Histograms in CNN architecture

In order to introduce generalisation into the convolutional network framework and utilise the whole OCT slice two new layers need to be developed: a masking layer and a histogram layer.

The aim of introducing these layers is so that the full image can be used in the network but within a convolutional network the size of each input must be the same. This is why the whole image cannot be used directly as the extracted slices are each of different sizes. To pad the image is possible but then each input image has a large percentage of redundant information which will skew training. In the proposed architecture it is necessary to include the histogram layer after convolutional layers; this allows the outputs to be fed into the fully connected

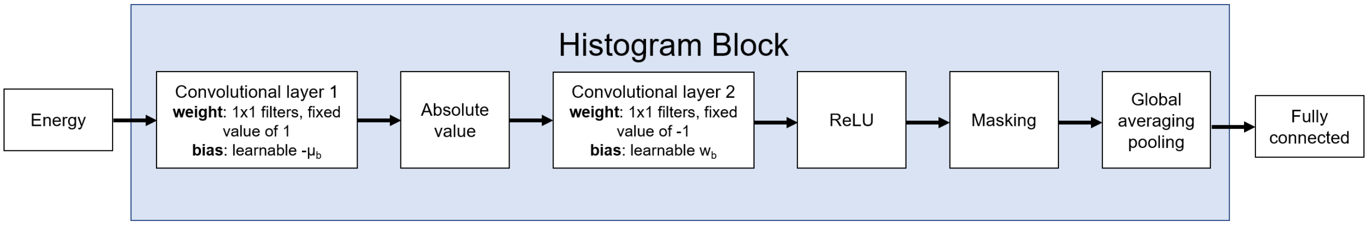


Figure 6.2: Combining a sequence of conventional layers to produce a histogram block.

layers. By introducing the masking layer before the histogram output we are able to remove all the redundant information introduced by the padding of the extracted region.

We aim to produce a histogram block, rather than a single histogram layer, in the convolutional network by combining and adapting a number of preexisting layers (see Fig. 6.2). The Equation 6.2 can be broken down into constituent parts:

$$x - \mu_b \tag{6.3}$$

$$|\ .\ | \tag{6.4}$$

$$w_b - . \tag{6.5}$$

$$max\{0, .\} \tag{6.6}$$

where the . represents the output of the preceding formula.

Each one of these can be implemented using an existing standard neural network layer. Equation 6.3 is analogous to a conventional convolutional layer with a fixed 1×1 filter with value 1, and a learnable vector of bias terms of b elements representing $-\mu$. For an input of $n \times m$, the output will be a $n \times m \times b$, effectively creating a value map for each of the b bins. The absolute value in Equation 6.4 can easily be implemented with the output the same dimension as the input. Equation 6.5 can also be implemented using a convolutional layer. In this instance the filters are of size 1×1 with a fixed value of -1. The bias terms are a vector of b elements representing w, where element b is the value of w_b. The maximum function can be implemented using a Rectified Linear Unit (ReLU). The final output from these layers is an $n \times m \times b$ set of maps. To convert this to a histogram a global average pooling layer is added. For each of the b value maps the average value is calculated. This results in a vector of b elements which is essentially a histogram of the inputs. By designing the histogram as a block of layers this makes the parameters μ_b and w_b optimisable allowing the histogram itself to be learnable.

In addition to the layers which make up the histogram block we need to add an extra two layers. For the histogram to be effective the maximum and minimum values of the input must be known so bin centres and bin widths can be initialised. The most efficient way to do this is to ensure all the input values are in the range $[0, 1]$. This can be simply achieved by introducing an energy layer which feeds into the first layer of the histogram block. We elect to use a sigmoid layer. The sigmoid function is a logistic function which always outputs a value between 0 and 1. It is calculated as:

$$S(z) = \frac{1}{1 + e^{-z}} \tag{6.7}$$

This type of function is commonly used in fully connected layers as an activation function to ensure values stay in a restricted range.

One of the innovations of our work is being able to use a convolutional architecture for inputs of variable size. To do this we need to introduce a masking layer. The inputs to the first layer of a convolutional network must all be of the same size and so each choroid is padded with zeros to make all inputs of consistent dimension. However, as we don't want to train our network on these redundant padded zeros we must introduce a masking layer. We place this layer within the histogram block before the global average pooling layer as this is the last stage where the layer dimensions are consistent with the input dimensions. The inputs to the masking layer are the outputs of the previous layer, in this case the ReLU layer of the histogram framework, and the original input image for the whole network. The input image is converted to a binary mask with a value for 1 if the pixel has a value, else zero. Each of the b layers of the output of the ReLU layer are multiplied by this mask. This removes any redundant values allowing only the area of the choroidal layers to contribute to learning the weights of the network. The output of this layer is the same size as the input and is passed to the global average pooling layer to produce the histogram.

6.4 Experimental Setup

Our proposed architecture combines a ResNet architecture with the histogram framework, as shown in Fig. 6.3. In Section 2.1 we introduced ResNet which is the current state-of-the-art deep learning technique for object recognition. We aim to utilise its feature extraction and classification abilities and combine with our histogram block to take advantage of the whole choroid.

In our implementation we are going to use the histogram ResNet in an end-to-end framework. The input images are fed into the network and the output is the slice classification. Based on other works [22, 28, 32] which found texture recognition can be achieved using only a small number of convolutional layers we employ a shallow ResNet architecture. We, as seen in Fig. 6.3, use a shallow network consisting of three residual blocks combined with the histogram block, initial pooling and convolutional layers, and fully connected layers feeding into the classification layer.

As all inputs need to be of the same dimension, each of the choroids are padded to make them all of the same size. This results in relatively large input image dimensions; for our dataset this was a set of inputs of 428×512 pixels. To reduce the spatial information, to decrease the computational expense and increase the speed of the network, we add initial pooling. For the main histogram branch we use a pool size of 2 before a convolutional layer which maintains value size and a second pooling layer with a pool size of 2. Simultaneously, we perform pooling on the original image with pool size 4; this branch will pass directly to the masking layer later in the network. Both branches have values of the same dimensions which is necessary for the masking layer. Pooling aggressively in the first few layers is used in ResNet in order to decrease spatial information and allow deeper networks and quicker computation. The convolutional layer mentioned consists of 64 filters of size 9 × 9, consistent with our previous methods. We normalise the output of the convolutional layer using a ReLU layer.

The output of the second pooling layer feeds into the first of three sequential residual blocks with convolutional layers with 16, 32 and 64 filters respectively, all of size 3 × 3. This is consistent with the style of architecture used in [10] in which small filters are used and the number of filters in each block increases as the network gets deeper.

The output of the final residual block is passed to the energy layer and the histogram block as discussed in Section 6.2. For the histogram 50 bins are used meaning the output of the histogram block is a vector of 50 elements. This vector is passed to a fully connected layer with 128 nodes before a fully connected layer with three nodes for the three classes. This allows the connections between the histogram structure and the classes to be learnt. A Softmax layer is used for error calculation to allow a probability distribution for each of the classes.

This will produce an individual classification for each slice. For leave-one-out-patient-out cross validation we then hold a majority vote of all the slice classifications of an eye to produce the classification for the patient.

6.5 Results

We tested the histogram ResNet on 2-fold validation and on leave-one-patient-out cross validation. We also tested using a ResNet architecture using the patch-based approach so as to be able to compare performance between the two methods.

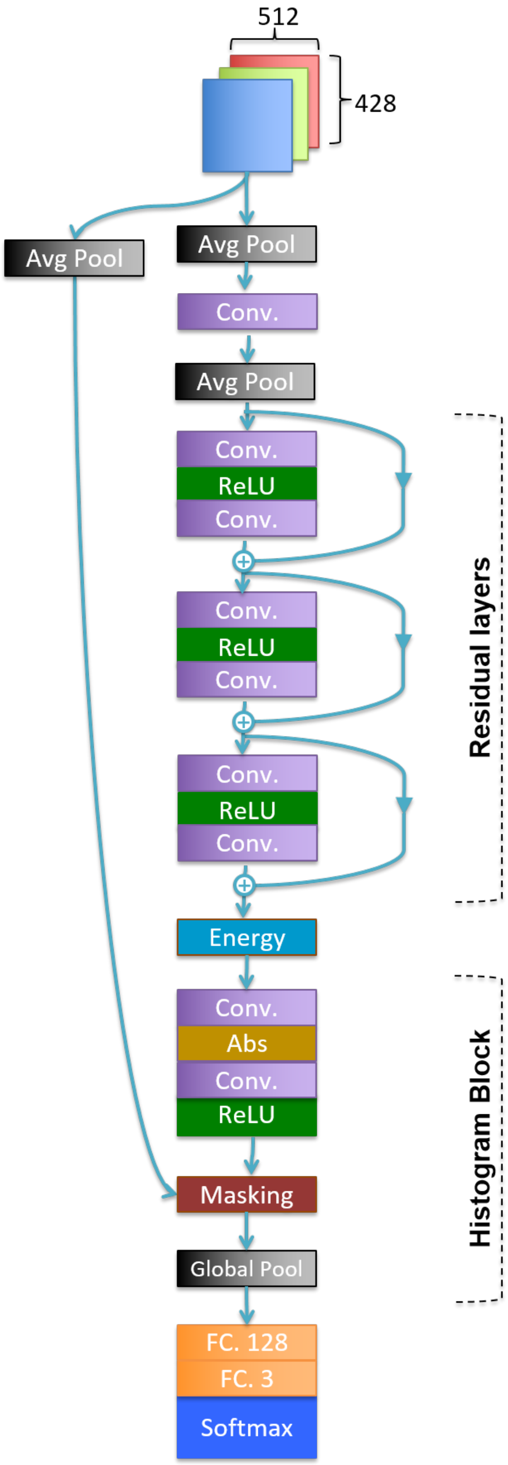


Figure 6.3: Architecture for ResNet architecture with embedded learnable histogram framework.

Tables 6.4 and 6.5 shows the results obtained using ResNet with the histogram architecture and using the path-based approach in an end-to-end framework respectively for 2-fold cross validation. The patch-based approach returns a per-slice accuracy of 43.3%. Using the histogram-based approach we achieved superior per-slice accuracies of 60.8%. This demonstrates the difficulty of directly using conventional deep learning techniques with a patch-based approach. As the input images are too small and each contain too little information about features they are unable to learn enough to produce accurate results. These results, however, were not as strong as those obtained in the approaches proposed in Chapter 5. As the input images were so large, 428×512 pixels, it was necessary to add pooling layers early on in the proposed network. Whilst this is a common technique in normal ResNets, the need for a masking layer may cause this to have a negative effect on performance. The image to be used for masking has also to be pooled before being converted into a binary image to act as a mask. The effect of pooling on this is to make the mask, and therefore the output from the masking layer, a less accurate representation of the region of interest. Examining Table 6.4, it is clear that the biggest difficulty is distinguishing between the Early AMD and Healthy classes. Pathologically, the features exhibited in Early AMD would be expected to be smaller than those in Wet AMD. This network struggles to identify these features of Early AMD which distinguish it from Healthy and this could be exaggerated by the effect of pooling. Prediction of Wet AMD is notably high, with 89% accurate classification of Wet AMD slices. Equally, the instances of Healthy or Early AMD slices being categorised into this class are low, at only 12.1% and 12.9% respectively. The network is able to accurately distinguish this class demonstrating its potential for discriminative classification. By using our embedded learnable histogram network we are able to utilise the learning abilities of ResNet across the whole of the choroidal image. Although the size of the used dataset is limited the improvement offered by this unique architecture is encouraging.

We are also interested in the per-patient accuracy so carry out leave-one-patient-out cross validation. In this, in each run one eye from each class was kept for testing while the classifiers were trained on the other 24 eyes from each class. As the convolutional network produces per-slice predictions we want to amalgamate the predictions from a single patient to produce an overall prediction. To achieve this we have a majority vote to produce a per-patient classification. Tables 6.1 and 6.2 show the results for leave-one-patient-out cross validation for per-slice and per-patient accuracies. Overall, per-eye prediction rates of 44.8% and 46.7% were achieved per-patient. A majority vote of the slice categorisations for each eye is used to calculate the per-patient classification. Table 6.3 shows the per-patch results of using ResNet for end-to-end classification for leave-one-patient-out cross validation. It is unable to identify the pertinent features needed to differentiate between the patches and resultantly classes all images into the control class. Consequently, per-slice and per-patient values are the same, 33.3%. This demonstrates the added difficulty of leave-one-out cross validation compared to 2-fold. We are working with a small dataset with only 75 patients and only 25 patients for each class. Furthermore, all slices within an eye are assigned the same categorisation as the same the eye overall. This may not necessarily be an entirely favourable method as not all slices will contain equal signs of their category and this may have an adverse effect on training and testing classification. With these considerations, the results achieved for leave-one-patient-out are promising.

Looking at the results we observe a number of points of interest indicative of the dataset. We notice that eyes which are classed as Wet are least likely to be predicted as Healthy (7.9% per-slice). Wet should show the most distinguishable features, whereas healthy eyes should show few, and so the two should have the most obvious differences. Similarly Wet eyes are most commonly correctly identified highlighting the prominence of perceptible features. We also note that eyes in the Early AMD class have the lowest rate of correct categorisation. They are at a stage between the other two classes and they may show features consistent with either of the other classes. As the disease is progressive, the features shown will be on a scale with some towards the healthy end whereas others may be closer to developing Wet AMD. Overall, the results seem consistent with the clinical assumptions that the similarities are greater between Healthy and Early, and Early and Wet than there are between Healthy and Wet. This cohesion with clinical understanding implies distinguishing features exist and are being identified, however, potentially due to the small size of the dataset, these features are not being picked up accurately in all slices.

	Healthy	**Early AMD**	**Wet AMD**	**Avg.**
Healthy	**37.8**	**38.8**	7.9	**44.8**
Early AMD	30.0	25.3	20.7	
Wet AMD	32.2	36.0	**70.5**	

Table 6.1: Confusion matrices of per-slice leave-one-patient-out cross validation for convolutional network with histogram framework (%)

	Healthy	**Early AMD**	**Wet AMD**	**Avg.**
Healthy	12.0	12.0	8.0	**46.7**
Early AMD	**44.0**	**44.0**	8.0	
Wet AMD	**44.0**	**44.0**	**84.0**	

Table 6.2: Confusion matrices of per-patient leave-one-patient-out cross validation for convolutional network with histogram framework (%)

	Healthy	**Early AMD**	**Wet AMD**	**Avg.**
Healthy	**100**	**100**	**100**	**33.3**
Early AMD	0	0	0	
Wet AMD	0	0	0	

Table 6.3: Confusion matrices of per-slice leave-one-patient-out cross validation for ResNet architecture (%)

	Healthy	Early AMD	Wet AMD	Avg.
Healthy	**55.0**	**48.7**	0.6	**60.8**
Early AMD	32.9	38.4	10.4	
Wet AMD	12.1	12.9	**89.0**	

Table 6.4: Confusion matrices of per-slice 2-fold cross validation for ResNet architecture with histogram framework (%)

	Healthy	Early AMD	Wet AMD	Avg.
Healthy	34.1	30.8	20.9	**43.3**
Early AMD	**36.7**	**47.0**	30.3	
Wet AMD	29.2	22.2	**48.8**	

Table 6.5: Confusion matrices of per-slice 2-fold cross validation for ResNet architecture (%)

Chapter 7

Conclusions and Future Work

7.1 Conclusions

Using a dataset of OCT scans of patients from three stages of AMD we explored the hypothesis that the texture of the choroidal region could be used for classification of these stages. This area has little research due to the recentness of accurate choroidal imaging. The dataset presented the challenge that the choroids were of varying size and shapes, where standard machine learning techniques require consistent input size. We presented two main approaches of dealing with this problem, producing promising results and introducing a novel architecture for deep learning.

7.1.1 Patch-based

The first approach involved extracting patches of consistent size from each of the choroidal layers. Using these we were able to exploit the feature-learning ability of CNNs but due to the limited size of the patches could not use them directly for accurate end-to-end classification. Instead, we extracted a bank of kernels from the first convolutional layer and used these as feature extractors. In traditional texture recognition techniques, hand-picked feature extractors such as

Gabor filters or wavelets are used for feature extraction with the resultant feature descriptor being passed directly onto machine learning classifiers. Utilising CNNs in this way allows us to use learnable feature extractors rather than hand-picked ones. This allows the development of a set of filters which are best suited to the data rather than having to make assumptions about what features exist and choosing filters accordingly. The learnt filters, when convolved across the input images produce a set of primitive features for the whole of the choroidal region.

At this stage the feature vectors are still of inconsistent length of the same size as each of the choroidal regions. Our first method of this approach employs using histograms at this stage; this allows feature vectors of consistent length to be produced and also enables generalisation. These can then be passed to supervised classifiers to produce predictions. The results demonstrate the feasibility of the method, where it outperforms solely using the feature learnt by the CNN, and promising quantitative results were reported.

We also presented a second method, in which we examine statistical and spatial information to develop a set of high-level features which can be used for accurate classification. From the primitive features statistical information is introduced by mining textons via clustering. The spatial arrangement of the features are examined using LBPs. Histograms are computed from the resultant output to produce the feature descriptors. These are passed onto the machine learning techniques for supervised classification. The accuracies achieved with this method outperform the previous method. The discriminative random forest classifier performed particularly well due to the high-level features developed.

Both methods, and the second in particular, produced higher accuracies than when using hand-crafted Gabor filters as the feature extraction method. Gabor filters have widely been shown to produce top results in texture recognition and medical imaging problems but their inferior performance demonstrates the benefits of the deep learning approach. By using self-learning techniques and not making assumptions about what features are expected to be found, a set of features can be developed which are more discriminative leading to better classification accuracies.

7.1.2 End-to-end Histogram Framework

In this thesis we have introduced a novel deep learning architecture for dealing with irregular shape input images and have produced promising results using a limited-sized dataset. We proposed a new approach for incorporating histograms into the framework of a convolutional network. A method by which the bin centre and width of the histogram could automatically be learnt was used. Introducing a masking layer allowed us to fully use a dataset comprised of irregularly shaped and sized images.

Our new histogram block was combined with state-of-the-art ResNet to produce a convolutional network architecture. This method produced promising results in leave-one-patient-out cross validation, a difficult task with a small dataset. It far outperformed the ResNet architecture used directly for classification with the patch-based approach for 2-fold cross validation. Given a larger dataset with more training data is reasonable to presume the results would further improve. This technique could have wide-ranging applications, allowing deep learning techniques to be used for datasets which are not uniform, particularly common in medical imaging where inter-patient variety is high.

7.2 Future Work

As with all research, beginning to explore a specific area leads to a set of branches developing on different directions the work could go and how it could be furthered.

One of the limiting factors in this research was the size of the dataset. Whilst results were promising the number of patients was limited affecting the overall accuracy and reliability of the classification results particularly for leave-one-patient-out classification. Working with temporal data within this field would also be an interesting research opportunity. By taking scans of the same set of patients' eyes at certain time intervals would show what changes occur as AMD progressed and machine learning could be utilised to predict which healthy patients are more likely to develop AMD and which of those in the early stages are more likely to develop the Wet form of the disease. As the disease is treatable with timely diagnosis, an ability to anticipate the likelihood of development is invaluable.

Moving away from the specific area we investigated, the validity of our methods could further be expanded through testing the techniques on other sets of data in other fields. In machine learning it is often the aim to have a generalisable model and so testing on a range of datasets is judicious.

For the histograms embedded within the CNNs there are a wide range of options which could further be explored in this area. We only tested a single architecture whereas various different ones could be tested. Furthermore, how the parameters within the histogram block, such as number of bins, affect performance was not fully explored due to time constraints. The data this method is tested on could also be expanded. The deep learning architecture is able to take inputs of different sizes and shapes and this could have applications in various other medical imaging areas where the regions of interest from the input images extracted may vary from patient to patient. Furthermore, this method could have wider applications in image recognition and classification. Currently input images for standard CNNs must be of consistent dimensions, or using multi-scaling must be one of a finite number of dimensions, meaning that cropping or patch extraction has to be used if images are not consistent in size. This method allows for the combination of datasets each tailored to a particular size of images. As accuracy achieved in this area relies heavily on the size of the dataset available for training and testing, by being able to amalgamate datasets this could have a beneficial effect on performance.

In this work we have demonstrated that the texture of the choroidal region can be used to classify stages of AMD and have presented methods which overcome the challenges posed by a dataset of irregularly shaped images.

Bibliography

[1] Y. Kanagasingam, A. Bhuiyan, M. D. Abramoff, R. T. Smith, L. Goldschmidt, and T. Y. Wong. Progress on retinal image analysis for age related macular degeneration. *Progress in retinal and eye research*, 38:20–42, 2014.

[2] James Loughman, John Nolan, James Stack, and Stephen Beatty. Online amd research study for optometrists: current practice in the republic of ireland and uk. 2011.

[3] Christoph W Spraul, Gabriele E Lang, Hans E Grossniklaus, and Gerhard K Lang. 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. *Survey of ophthalmology*, 44:S10–S32, 1999.

[4] Frank Rosenblatt. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386, 1958.

[5] Yann LeCun and Yoshua Bengio. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, 3361(10):1995, 1995.

[6] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bern-

stein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[7] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[8] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[9] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[11] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[12] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv preprint arXiv:1606.00915*, 2016.

[13] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016.

[14] Kevis-Kokitsi Maninis, Jordi Pont-Tuset, Pablo Arbeláez, and Luc Van Gool. Convolutional oriented boundaries. In *European Conference on Computer Vision*, pages 580–596. Springer, 2016.

[15] Xianghua Xie. A review of recent advances in surface defect detection using texture analysis techniques. *ELCVIA: electronic letters on computer vision and image analysis*, 7(3):1–22, 2008.

[16] G Castellano, L Bonilha, LM Li, and F Cendes. Texture analysis of medical images. *Clinical radiology*, 59(12):1061–1069, 2004.

[17] Zhenhua Guo, Lei Zhang, and David Zhang. A completed modeling of local binary pattern operator for texture classification. *IEEE Transactions on Image Processing*, 19(6):1657–1663, 2010.

[18] George R Cross and Anil K Jain. Markov random field texture models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (1):25–39, 1983.

[19] Dennis Dunn and William E Higgins. Optimal gabor filters for texture segmentation. *Image Processing, IEEE Transactions on*, 4(7):947–964, 1995.

[20] Scott Doyle, Michael Feldman, John Tomaszewski, and Anant Madabhushi. A boosted bayesian multiresolution classifier for prostate cancer detection from digitized needle biopsies. *Biomedical Engineering, IEEE Transactions on*, 59(5):1205–1218, 2012.

[21] Joao VB Soares, Jorge JG Leandro, Roberto M Cesar Jr, Herbert F Jelinek, and Michael J Cree. Retinal vessel segmentation using the 2-d gabor wavelet and supervised classification. *Medical Imaging, IEEE Transactions on*, 25(9):1214–1222, 2006.

[22] Fok Hing Chi Tivive and Abdesselam Bouzerdoum. Texture classification using convolutional neural networks. In *TENCON 2006. 2006 IEEE Region 10 Conference*, pages 1–4. IEEE, 2006.

[23] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613, 2014.

[24] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

[25] Vincent Andrearczyk and Paul F Whelan. Using filter banks in convolutional neural networks for texture classification. *Pattern Recognition Letters*, 84:63–69, 2016.

[26] Vincent Andrearczyk and Paul F Whelan. Deep learning for biomedical texture image analysis. In *Irish Machine Vision & Image Processing Conference proceedings IMVIP*, volume 2016, 2016.

[27] Mircea Cimpoi, Subhransu Maji, and Andrea Vedaldi. Deep filter banks for texture recognition and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3828–3836, 2015.

[28] Vincent Andrearczyk and Paul F Whelan. Texture segmentation with fully convolutional networks. *arXiv preprint arXiv:1703.05230*, 2017.

[29] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015.

[30] Leon Gatys, Alexander S Ecker, and Matthias Bethge. Texture synthesis using convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 262–270, 2015.

[31] Javier Portilla and Eero P Simoncelli. A parametric texture model based on joint statistics of complex wavelet coefficients. *International journal of computer vision*, 40(1):49–70, 2000.

[32] Ivan Ustyuzhaninov, Wieland Brendel, Leon A Gatys, and Matthias Bethge. Texture synthesis using shallow convolutional networks with random filters. *arXiv preprint arXiv:1606.00021*, 2016.

[33] Feng Yang, Gui-Song Xia, Liangpei Zhang, and Xin Huang. Stationary dynamic texture synthesis using convolutional neural networks. In *Signal Processing (ICSP), 2016 IEEE 13th International Conference on*, pages 1135–1139. IEEE, 2016.

[34] Du Tran, Lubomir Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 4489–4497, 2015.

[35] Luiz G Hafemann, Luiz S Oliveira, Paulo R Cavalin, and Robert Sabourin. Transfer learning between texture classification tasks using convolutional neural networks. In *Neural Networks (IJCNN), 2015 International Joint Conference on*, pages 1–7. IEEE, 2015.

[36] Christoph W Spraul, Gabriele E Lang, and Hans E Grossniklaus. Morphometric analysis of the choroid, bruch's membrane, and retinal pigment epithelium in eyes with age-related macular degeneration. *Investigative ophthalmology & visual science*, 37(13):2724–2735, 1996.

[37] Andreea Gheorghe, Labib Mahdi, and Ovidiu Musat. Age-related macular degeneration. *Romanian Journal of Ophthalmology*, 59(2):74–77, 2015.

[38] S. Mrejen and R. Spaide. Optical coherence tomography: Imaging of the choroid and beyond. *Survey of Ophthalmology*, 58(5):387–429, 2013.

[39] B. Povazay, K. Bizheva, B. Hermann, A. Unterhuber, H. Sattmann, A. Fercher, W. Drexler, C. Schubert, P. Ahnelt, M. Mei, R. Holzwarth, W. Wadsworth, J. Knight, and P. Russell. Enhanced visualization of choroidal vessels using ultrahigh resolution ophthalmic OCT at 1050 nm. *Opt. Express*, 11(17):1980, 2003.

[40] W. Drexler, U. Morgner, R. K. Ghanta, F. X. Kartner, J. S. Schuman, and J. G. Fujimoto. Ultrahigh-resolution ophthalmic optical coherence tomography. *Nature Medicine*, 7(4):502–507, 2001.

[41] R. Priya and P. Aruna. Automated diagnosis of age-related macular degeneration from color retinal fundus images. In Electronics Computer, editor, *Technology (ICECT), 3rd International Conference on*, pages 227–230, IEEE, 2011. 2.

[42] Jathurong Sugmk, Supapom Kiattisin, and Adisom Leelasantitham. Automated classification between age-related macular degeneration and diabetic macular edema in oct image using image segmentation. In *Biomedical Engineering International Conference (BMEiCON), 2014 7th*, pages 1–4. IEEE, 2014.

[43] Sina Farsiu, Stephanie J Chiu, Rachelle V O'Connell, Francisco A Folgar, Eric Yuan, Joseph A Izatt, Cynthia A Toth, et al. Quantitative classification of eyes with and without intermediate age-related macular degeneration using optical coherence tomography. *Ophthalmology*, 121(1):162–172, 2014.

[44] R. Koprowski, S. Teper, Z. Wróbel, and E. Wylegala. Automatic analysis of selected choroidal diseases in oct images of the eye fundus. *BioMedical Engineering OnLine*, 12:117, 2013.

[45] Louise Terry, Nicola Cassels, Kelly Lu, Jennifer H Acton, Tom H Margrain, Rachel V North, James Fergusson, Nick White, and Ashley Wood. Auto-

mated retinal layer segmentation using spectral domain optical coherence tomography: Evaluation of inter-session repeatability and agreement between devices. *PloS one*, 11(9):e0162001, 2016.

[46] Age-Related Eye Disease Study Research Group and others. The age-related eye disease study system for classifying age-related macular degeneration from stereoscopic color fundus photographs: the age-related eye disease study report number 6. *American journal of ophthalmology*, 132(5):668–681, 2001.

[47] Ehab Essa, Xianghua Xie, Igor Sazonov, Perumal Nithiarasu, and Dave Smith. Shape prior model for media-adventitia border segmentation in ivus using graph cut. In *International MICCAI Workshop on Medical Computer Vision*, pages 114–123. Springer, 2012.

[48] Jonathan-Lee Jones, Xianghua Xie, and Ehab Essa. Combining region-based and imprecise boundary-based cues for interactive medical image segmentation. *International journal for numerical methods in biomedical engineering*, 30(12):1649–1666, 2014.

[49] Ehab Essa, Xianghua Xie, and Jonathan-Lee Jones. Minimum s-excess graph for segmenting and tracking multiple borders with hmm. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 28–35. Springer, 2015.

[50] M Pakdel and F Tajeripour. Texture classification using optimal gabor filters. In *Computer and Knowledge Engineering (ICCKE), 2011 1st International eConference on*, pages 208–213. IEEE, 2011.

[51] Simona E Grigorescu, Nicolai Petkov, and Peter Kruizinga. Comparison of texture features based on gabor filters. *Image Processing, IEEE Transactions on*, 11(10):1160–1167, 2002.

[52] Yoshinori Hayashi, Toshiaki Nakagawa, Yuji Hatanaka, Akira Aoyama, Masakatsu Kakogawa, Takeshi Hara, Hiroshi Fujita, and Tetsuya Yamamoto. Detection of retinal nerve fiber layer defects in retinal fundus images using gabor filtering. In *Medical Imaging*, pages 65142Z–65142Z. International Society for Optics and Photonics, 2007.

[53] Anil K Jain and Farshid Farrokhnia. Unsupervised texture segmentation using gabor filters. pages 14–19, 1990.

[54] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[55] Anna Bosch, Andrew Zisserman, and Xavier Munoz. Image classification using random forests and ferns. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.

[56] Manik Varma and Andrew Zisserman. A statistical approach to texture classification from single images. *IJCV*, 62(1):61–81, 2005.

[57] Timo Ojala, Matti Pietikäinen, and David Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.

[58] Zhe Wang, Hongsheng Li, Wanli Ouyang, and Xiaogang Wang. Learnable histogram: Statistical context features for deep neural networks. In *European Conference on Computer Vision*, pages 246–262. Springer, 2016.

List of Figures

List of Tables