A frequency based method for Retinal Nerve Fiber Layer detection

Supervisor:
Prof. Alfredo Ruggeri

Co-Supervisor:
Prof. Emanuele Trucco

Student:
Alessia Antelmi

Master Degree in Bioengineering
21/02/2017 Academic Year 2016/2017

To all the people who keep
believing in me.

I want to deeply thank my family, my boyfriend and my friends for their trust in me. I also want to thank Prof. Emanuele Trucco and the whole CVIP group at University of Dundee, Prof. Alfredo Ruggeri and all the professors who have accompanied me in this path.

Abstract

Retinal Nerve Fiber Layer is one of the layers that make up retina. Its detection has been recently taken into consideration because its visibility is used as a biomarker for a brain disease called dementia. The method developed in this thesis for RNFL detection works in the frequency domain, exploiting the rotation invariance of the circular integral of the magnitude of the Fourier Transform, applied at first to synthetic images, then to real retinal images. Achievements are not considered good for the current application.

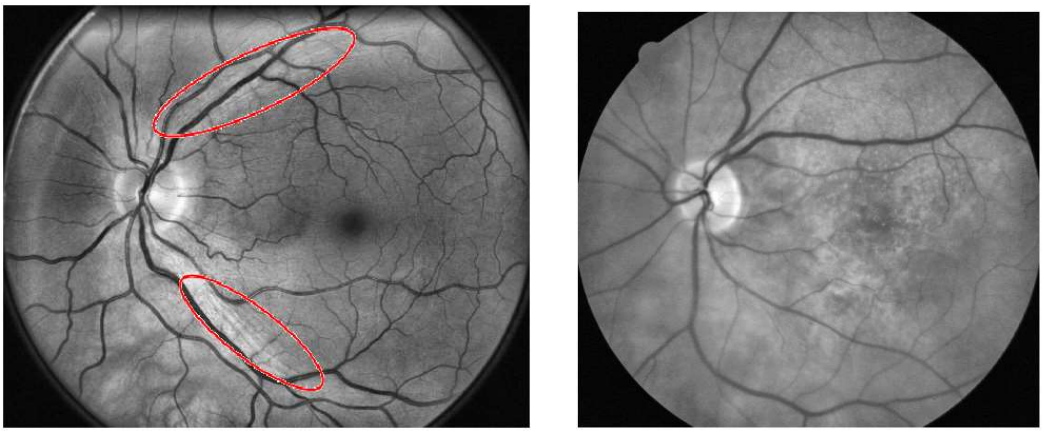
Contents

List of Figures

Chapter 1

Introduction

The work developed in this thesis has been done in the VAMPIRE group at the Computing School, University of Dundee (UK), where the attention is focused on Computer Vision and Image Processing. "VAMPIRE" stands for Vascular Assessment and Measurement Platform for Images of the REtina and is aimed to quantify retinal vessels properties by using Fundus Camera images.
Specifically, this thesis regards the analysis of retinal images coming from a branch of imaging called *fundus imaging*: its aim is the detection of some peculiar patterns over the retinal images, called *Retinal Nerve Fiber Layer*, that actually are unmyelinated axons of the ganglion cells and usually lie among blood vessels towards the optic nerve. RNFL analysis is part of a bigger project called *Multi-modal retinal biomarkers for vascular dementia*, arisen by a collaboration between Dundee and Edinburgh universities.
The interest in the extraction of such patterns derives from their visibility: Figure 1.1 shows indeed that images can be distinguished into two classes, according to the visibility or invisibility of RNFL. This is important since it has been proved a relation between these stripy patterns and the presence of a brain disease, called *dementia*. Dementia is characterized by symptoms, e.g. memory loss, that can gradually lead to severe problems in activities like thinking or talking and then can get life more complicated.
Detection of ripple patterns is matter of texture analysis, where texture can be generally defined as the repetition of one or more elements, like textile threads, rocks in a river or, in this case, RNFL in fundus camera images.



(a) Visible RNFL. (b) Invisible RNFL.

Figure 1.1: Retinal images examples.

Detection is here done in the frequencies domain through a method deriving from a work done in 2000, whose title is *Detecting Ripple Patterns in Mission Videos* [27], since it shows invariance to rotation. The main idea of the method is the calculation of the circular integral of the Fourier Transform magnitude for a variety of radius values. Hence, the correspondent plot is inspected to look for its peaks.

At first this method has been tried for synthetic images, namely binary images described by a sinusoidal trend, for different frequencies and rotation's degrees and for an increasing density of noise. From this first stage, results were considered a good achievement since they were characterized by the expected property of rotation invariance. After that, 884 retinal images (given by CVIP group, Computing School, University of Dundee, UK) were analysed as well. Each image is divided into several square windows in order to deeply examine the potential presence of RNFL; hence, the integral of the magnitude is calculated for each window and results are observed. For real images this method doesn't seem to fit, since all images, both with visible and invisible RNFL, are classified as containing the pattern for the chosen threshold which divides the positive images from the negative ones. The attempt has been done in frequency domain in order to find a faster way to detect RNFL than the methods used in CVIP group for the same aim, like Scale Invariant Feature Transform or Local Binary Patterns; unfortunately, as expected, the frequency based method tried worsens the performances or, precisely, isn't adequate to the current application.

This thesis develops in six chapters, besides the current one. In Chapter 2, all the background knowledge is given: what the human eye is and which kind of imaging technique is used; then, Chapter 3 cites some works of literature that analysed stripy patterns or particular motifs in images. Chapter 4 explains both the methods used at the CVIP group for RNFL detection and two frequency based methods; Chapter 5 concerns the chosen method, the mathematical tools useful for this aim and the steps to follow; Chapter 6 shows the results obtained in synthetic images and in retinal images, hence Chapter 7 concludes the thesis with some further observations and ideas for some other works.

Chapter 2

Background knowledge

This thesis focuses on the anatomic part of the visual apparatus, the human eye, whose structure can be seen in Figure 2.1.

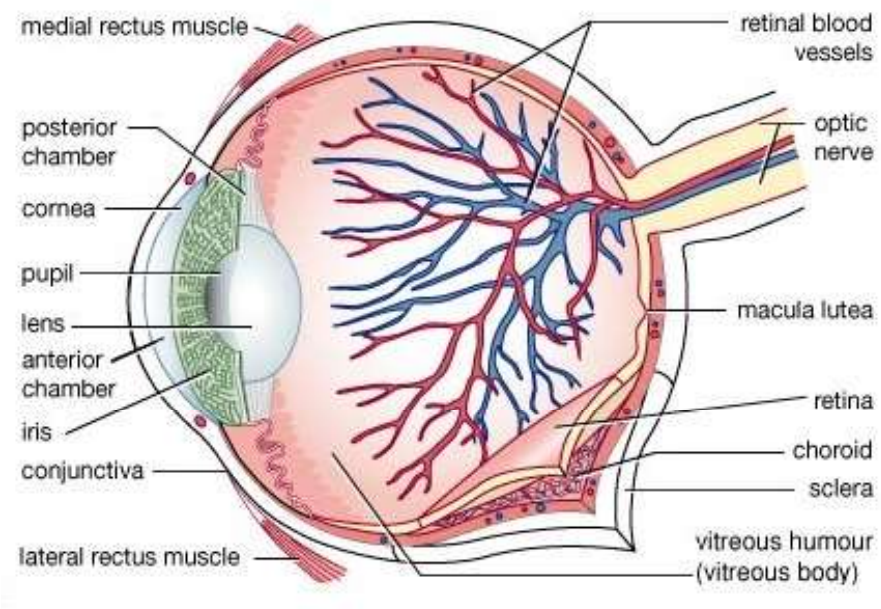
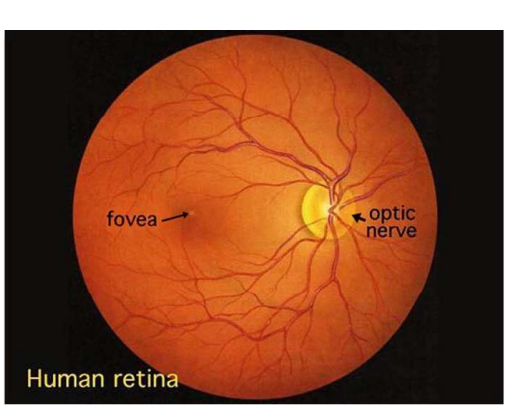


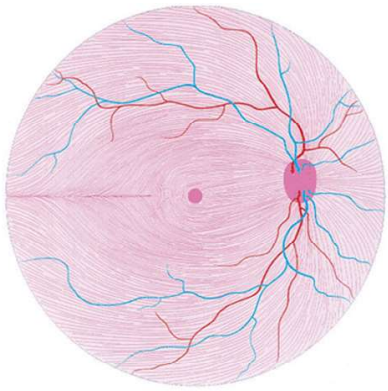
Figure 2.1: Structure of a human eye [6].

The ocular bulb, whose shape is spheroidal, is characterized by an anterior and a posterior pole, which are both covered by a white membrane, named *sclera*. The first mentioned pole, on the left part of Figure 2.1, is externally delimited also by a transparent lens, called *cornea*, which is made by connective cells; behind it, an anterior chamber is fulfilled by the *aqueous humour*, a fluid that keeps the ocular

globe steady. Within the eye, there is also the *iris*, a tissue that gives the specific visible colour to the eye; at its centre there is a *pupil*, which is dark since the light passing through the eye isn't reflected back. At the posterior chamber level, there is a biconvex lens, called *crystalline lens*, which is responsible for the focal distance adjustment. Then, the eyeball is completely filled by a jelly substance, the *vitreous body*, whose function is to press retina against choroid. *Retina* is a membrane wrapping internally the eyeball which allows the eyes to see in conditions of light and darkness through its photosensitive cells: *cones*, responsible for the photopic vision and the details and colours perception, and *rods*, responsible for the scotopic vision, more sensible to light and much more numerous than cones (130 million versus 7 million); at the end, *choroid*, a membrane localized between retina and sclera, is aimed at bringing blood in the eye.
The posterior pole, on the right part of Figure 2.1, is characterized by the presence of the *optic disk*, from which the optic nerve begins. The *optic nerve* carries the sensory nerve impulses from the cells located in the retina to the brain and contains the blood vessels that vascularize the retinal membrane.
The ophthalmoscope is an instrument used to brighten the lining of the eye through the pupil in order to visualize the blood vessels that feed retina with blood, obtaining the images studied in this thesis, an example of which can be observed in Figure 2.2a.



(a) An example of a fundus image [7].



(b) Scheme of Retinal Nerve Fiber Layer [7].

Figure 2.2: A real and an ideal retinal images.

The *fovea* is located inside a region called *macula* by ophthalmologists, which is situated in the posterior pole of the eye, on the retinal membrane. The fovea is essential for an accurate vision: in its central part it is rich of cones, whereas towards its edges it is rich of rods; this behaviour is opposite to the one of the optic nerve, since the last one starts in a point called *blind spot*, i.e. a point which has no cones nor rods.

At the end, the imaging technique deriving from the use of the ophtalmoscope, called *fundus imaging*, is aimed at representing image intensities as quantities of reflected light in bidimensional images. There exist several *fundus imaging* modalities [8], among which there are *fundus photography*, where intensities are related to a specific waveband, and *color fundus photography*, where intensities are the representation of the quantity of Red, Green and Blue wavebands. Fundus Camera imaging has some challenging issues: 1) the size of pupil is limited to a few millimetres and light passing through it has to illuminate the retina; 2) illumination and imaging beams must be used in two separate ways in order to not worsen image contrast. The forementioned problems lead to high costs and the need of make FC imaging more accessible and cheaper. *Fundus photography* is largely used for the detection of some pathologies, e.g. diabetic retinopathy, glaucoma and *macula* degeneration related to patients' age.

A newer technique spread in retinal imaging is *Optical Coherence Tomography* (OCT), which estimates the depth where a particular back-scatter starts; it's important to choose a specific backscatter since they come from the variability of the refractive index among tissues and the deeper the origin is, the more time is needed by the backscatter to come back to the sensor. In this case, the appropriate instrument is the *White Light Interferometer*, characterized by a wave length longer than the visible wave in order to obtain autocorrelation for a brief time gap. *OCT* is commonly used for the diagnosis and the control of patients affected by macula degeneration, diabetic retinopathy and inflammatory retinal diseases; it is patient-friendly since it works in a non-invasive way and the image acquisition lasts only few seconds.

Retinal Nerve Fiber Layer, one of the layers that make up retina, is covered by ganglion cell axons that are directed to the optic nerve in a bent form, as the scheme in Figure 2.2b shows. RNFL thickness has been recently studied by OCT

in order to find an association of its presence to some pathologies and it has been proved to be a biomarker for dementia [29]. The visibility of RNFL in FC is not always clear and it is indeed used as a marker: if RNFL is classified as visible, it is proposed as a biomarker.
From this Chapter, it's hence understandable the interest in the analysis of retinal images.

Chapter 3

Related work

Detection of ripple patterns is a field which has been studying in the last years in several different applications.
G. Baciu and Y. Cai [9] proposed a detection method in near regular texture images, based on the *texton* definition. A texton is a repetitive pattern in a texture, associated to a geometric (linear) transformation which makes it match a specific area of the texture region. Thus, in this particular type of images, a patch differs from another one just because of the deformation introduced by the transformation; the main idea to find the repetitive pattern is to compose an image patch T with a geometric transformation and then measure the similarity between T and the composition just calculated. Then, a patch shape alignment is done to detect a global template to be used in the final phase of matching.
This work has then been improved by the same authors two years later [11]. The detection of the patches is based on the region-growing segmentation algorithm, first grouping local image patches into clusters and then jointly aligning them for matching. This is followed by an inference step that explains how the patterns previously found can be organised in a specific global geometric structure.
These methods can not be used in ripple patterns extraction since they are more suitable for the detection of elements like rings, grids and so on.
It's worth stressing the two major categories of texture analysis' methods [14]:

- *Structural Approach*: it's based on the definition of the texture as a repetition of some primitives, which are hard to find in natural images. The traditional Fourier Transform is hence applied to try to solve the problem.

- *Statistical Approach*: it's used to describe the stochastic properties of grey levels' spatial distribution in an image. The Grey Level Co-occurrence Matrix (GLCM) [15] usually fits for this aim since it counts how many times a pixel characterised by a specific intensity i occurs either vertically, horizontally or diagonally to adjacent pixels with intensity k. Then, if the image has N grey levels, the GLCM has size NxN and it is filled with the number of the occurrences of *(i,k)* in a chosen direction.
  Another statistical method is explained in [13] and it exploits the definition of texture unit and texture spectrum [14]. Actually, while a unit is made by the eight pixels surrounding a given pixel i, a spectrum is the frequency histogram of all the texture units. In this sense, the fuzzy texture unit can be obtained considering the characteristics of image's homogeneity through a similarity function. Then, some specific classes are set according to the probabilities calculated starting from the fuzzy texture unit of training images (with stripes, with grains, non-linear or homogeneous). At the end, classification of testing images is done and it shows an improvement in accuracy and robustness of the spectrum calculus.

A possible approach for stripe feature extraction consists in the use of morphological operators followed by Hough Transform, in the case of scrap paper images mosaics [12]. Scrap papers are put on a black background and scanned, to obtain the original images. Then, they are converted into grey-levels images and a *closing* operation[1] is done in order to extract the background and obtain binary images, where the white area is the scrap paper. After that, stripes are detected by means of a difference between the original scrap image and the binary image, and edge detection is carried out. At the end, Hough Transform[2] is used since it's able to detect lines in an accurate way.
Some works concern instead the extraction of symmetrical motif with the aim of improve methods relevant to real images, which are corrupted by illumination

[1]Closing is a spatial operator which, in the case of binary images, tends to smooth contours, fulfilling their holes and discontinuities, to merge objects wrongly detached and to remove noisy pixels from the background. In the case of grey-levels images, it removes small and dark details, avoiding variations of bright details. For further explanations, see [4], Chapter 9.

[2]Hough Transform is a global operator that converts all points of an image (generally, binary) into points in a new space to find the subsets of points belonging to straight lines, circles or any kind of shape. For further details, see [4], pp.755-760.

changes, background noise and deformation, just to mention some problems existing. The work [16] illustrates a new method that brings to a good behaviour in terms of motif detection; particularly, it is split into two phases: the extraction of the motif through the use of an autocorrelation function and the correction of the symmetrical motif with the centre of symmetry, calculated using particular features, called *Scale Invariant Feature Transform* (see Chapter 4).
In closing, this thesis has its basis on the studies done at University of Dundee regarding the classification and the localization of RNFL. In this sense, in a work among the related ones [28], some of the methods described in Chapter 4 were used to classify cell and specimen images; in particular, features were firstly extracted by means of Local Patterns, a variant of SIFT called Root-SIFT, Random Projection and Intensity Histograms and then encoded through four different kinds of methods, among which there was *Bag-of-Words*, explained at the beginning of Chapter 4 too. SIFT method gave better performances if compared to the other methods in the case of cell classification and in the case of small patches in specimen classification; for this reason, Scale Invariant Feature Transform was used for Retinal Nerve Fiber Layer detection in comparison with Local Directional Pattern, Random Projection and Filter banks. The good results obtained via SIFT features in terms of correct classification made possible the introduction of an automated system in [29] which is characterized by a better agreement with the clinicians than the methods of the State-of-the-Art. For RNFL's properties, this technique set a certain number of sub-categories that can stand for different variations of the pattern; then, a set of classifiers for the sub-categories is learnt. The margin between the set of visible RNFL (*positive bag*) and the set of invisible RNFL (*negative bag*) can hence be found in order to make predictions for each bag. The described procedure is called *Multiple Instance Learning*.

Chapter 4

Features Extraction

An important step in Image Analysis is the extraction of features. Features make possible the recognition and the description of objects in images in order to distinguish different objects' instances.
In Retinal Nerve Fiber Layer images, Bag of Features approach is used in order to classify categories. An essential point is the prior information that some images belong to a given class meanwhile others don't; according to this, it's possible to classify an element in a defined class taking into account the possibility of variations in terms of scale, rotation, translation and so on. In a more specific way, after the extraction of regions of a set of images, descriptors are computed to better characterize objects in the images and to make possible the discrimination of features in different images. Moreover, these descriptors are found with the idea of being invariant to little differences in the objects.
This first stage is followed by quantization that brings to cluster every object in the correct class through a chosen method. An example could be the K-means clustering that at first associates randomly each element to one of the k possible clusters and calculates every cluster's mean; then, it measures the distance of each element from the k means and eventually moves elements to another cluster to minimise the cost function.
Then, each image is represented by a frequency histogram in order to go through classification, that can be linear or non-linear.
On one hand, Chapter 4 aims at introducing and describing the basis of the techniques used in [28], [29] in order to understand the peculiarities of such methods.

On the other hand, it explains two techniques that work in the frequency domain, called *Fourier-based Rotation Invariant Feature* and *Local Binary Pattern Histogram Fourier Features* (see §4.2 and §4.3.2 respectively), because they have been taken into consideration for the choice of the method to implement throughout the thesis.

4.1 Scale Invariant Feature Transform

Scale Invariant Feature Transform (SIFT) method [19] works locally and so it is robust to occlusion and clutter. It gives as an output a set of feature vectors which are characterized by invariance to:

- translation;
- rotation;
- scale;
- illumination changes (even if partially)
- affine/3D projections (even if partially too).

First of all, rotation invariance in the scale space is achieved by building an image pyramid (See 4.1.1) in order to calculate maxima and minima of the difference of Gaussian in the bidimensional space. A comparison between each pixel and its neighbourhood is computed at the same pyramid level: if there's a maximum or a minimum, the nearest pixel location is searched in the next lowest pyramid's level, otherwise the research is done in the level above. In this way, several pixels are removed in a brief time; the remaining N points generate N feature vectors, called SIFT keys.
Each image's level is then divided into small square windows in order to compute magnitudes and orientations; after that, the histogram of local image gradient orientations is summed up. Thus, key locations are assigned to the obtained specific orientations to avoid dependence to lighting changes.
The features obtained by SIFT method are highly efficient and are extendible to a large range of different features; furthermore, they can be produced for both big and small objects and be used in a wide database of elements.

4.1.1 Image Pyramids

Image pyramids' method is a way for representing an image at different resolutions. In fact, a pyramid is made up by the same image decreasing in both resolution and size: this means that, at each step, we can imagine to apply a low-pass filter and we accept a loss of details from the original image.
For the sake of clearness, I report an easy example. Let's assume to have a N-by-N image. At the first step, a low-pass filter (in the best case, a Gaussian one) is applied just to worsen resolution, without any change in size; then, in the filtered image the downscaling is done, yielding to a (N/2)-by-(N/2) image. This procedure is repeated until the desired size and resolution are obtained. As shown in Figure 4.2, the loss of details obviously becomes more and more considerable going on with the downscaling.
In terms of object detection, it has been proved that convolving several copies of the object at increasing scales with the original image brings to the same result of convolving the object with several copies of the image at the decreasing resolution corresponding to the defined scales. It's usually chosen the second method for its efficiency and computational complexity.

4.2 Fourier-based Rotation Invariant Features

This technique [26] achieves invariance to rotation, using both the phase and the magnitude components of the Fourier Transform.
In fact, the image patch in Cartesian frame is firstly converted in polar coordinates in order to translate any kind of rotation into a circular shift. Thus, the Discrete Fourier Transform can be used for signals that diverge only by a circular shift and gives two signals differing only by a scale factor; every original signal can be written in terms of Fourier Transform magnitude and phase, i.e. if f(n) is the original signal of size N and $F_k = DFT[f(n)]$ its transform, f(n) is written as:

$$f(n) = \frac{1}{N}F_0 + \frac{1}{N}\sum_{k=1}^{N-1} |F_k| \cdot e^{j\cdot(\angle F_k + 2\pi\frac{kn}{N})}.$$

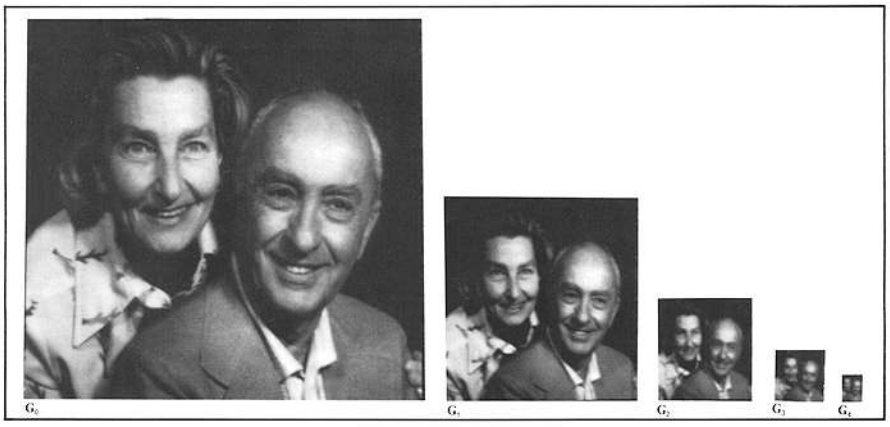


Figure 4.1: Image pyramid from [20].

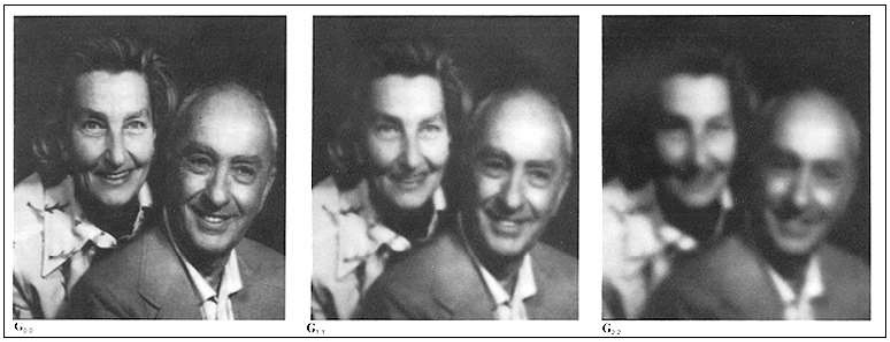


Figure 4.2: Effects only of the low-pass filtering [20].

It is also possible to keep information about the spatial relationship among pixels through rows entanglement, a process that allows shifts if the original image is uniformly rotated.

4.3 Local Binary Patterns

As the name suggests, Local Binary Patterns (LBP) [21] method works locally. LBP is a texture operator which, originally, makes a threshold of the 3-by-3 neighbourhood of each pixel p of interest in an easy way. If q is a neighbour pixel of p, the corresponding element of the binary mask b is set according to the following scheme:

$$b = \begin{cases} 1 & \text{if } q \geq p \\ 0 & \text{otherwise.} \end{cases} \tag{4.1}$$

In this way, the binary pattern for the pixel p can be obtained.
The threshold matrix is then multiplied, element by element, by a matrix of the same size containing weights in the eight pixels of p's neighbourhood: the sum of these products is LBP. For the case of Figure 4.3, the binary pattern is 01111010 and the Local Binary Pattern is $1 + 2 + 8 + 64 + 128 = 203$.

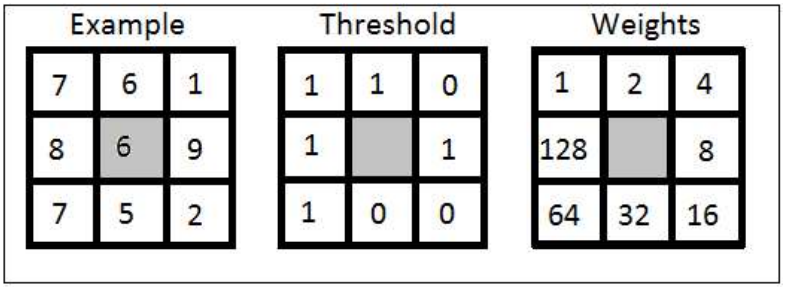


Figure 4.3: An example of the matrices used in Local Binary Pattern.

Despite its simplicity, LBP has the important property to be invariant to monotonic grey level changes.
The neighbourhood reported in Figure 4.3 is square but it can also be of other shapes. For example, if the pattern is circular, we can have the case of uniform/not uniform patterns. Uniform patterns are a particular kind of Local Binary Patterns which are characterized by at most two bitwise transitions from 1 to 0 and from 0 to 1.
A wide variety of Local Binary Pattern method exists; Local Directional Pattern and Local Binary Pattern Histogram Fourier Features are shown in the following subsections.

4.3.1 Local Directional Pattern

Local Directional Pattern (LDP) [23] is a stabler variant than the Local Binary Pattern method since it considers the edge response values in all the different directions. This means that LDP pattern code doesn't change if there's an addition of noise in the image, providing a greater consistency and robustness.
LDP assigns a 8-bit binary code to each pixel of an input image comparing the edge response of a pixel in different directions. In [23], eight directional edge response values are calculated for each pixel by means of eight 3-by-3 Kirsch masks in eight

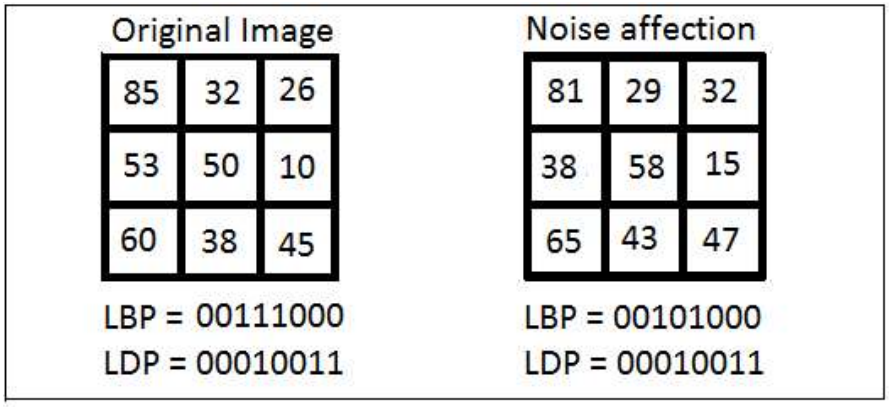


Figure 4.4: Original image's and noisy image's codes.

different orientations, reported for completeness:

$$\begin{bmatrix} -3 & -3 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & 5 \end{bmatrix}, \quad \begin{bmatrix} -3 & 5 & 5 \\ -3 & 0 & 5 \\ -3 & -3 & -3 \end{bmatrix}, \quad \begin{bmatrix} 5 & 5 & 5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}, \quad \begin{bmatrix} 5 & 5 & -3 \\ 5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix},$$

$$\begin{bmatrix} 5 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & -3 & -3 \end{bmatrix}, \quad \begin{bmatrix} -3 & -3 & -3 \\ 5 & 0 & -3 \\ 5 & 5 & -3 \end{bmatrix}, \quad \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ 5 & 5 & 5 \end{bmatrix}, \quad \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & 5 \\ -3 & 5 & 5 \end{bmatrix}.$$

The edge response can be extremely different from one pixel to another: if there's an edge or a corner, the response is higher in specific directions. The k principal directions are searched in order to write the LDP code: the k higher response values are set to 1 and, consequently, the other values are set to 0.
In this way, the LDP code is invariant to illumination changes and to noise, becoming preferable to LBP's, as the following example shows (Figure 4.4). LDP code indeed doesn't change if the image is affected by noise, while one bit of the LBP code has changed: this result stresses that gradients are more robust than grey level values.

4.3.2 Local Binary Pattern Histogram Fourier Features

Local Binary Pattern Histogram Fourier Features method (LBP-HF) [22] has its basis on the uniform Local Binary Pattern, characterised by a circular bit pattern

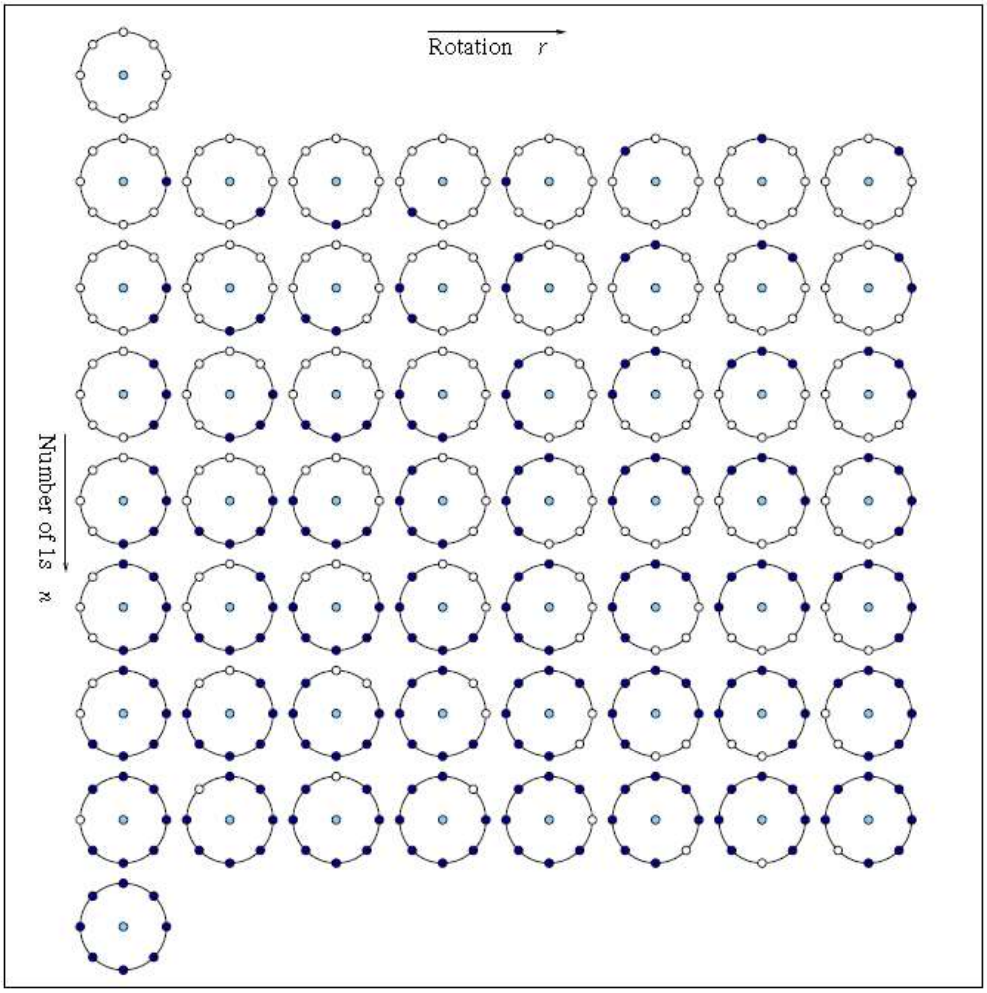


Figure 4.5: The uniform patterns in the case of 8 neighbours [22].

as mentioned in §4.3. According to this, if the neighbourhood has size 8, there can be 58 different uniform patterns, thanks to rotation (Figure 4.5). The Discrete Fourier Transform is applied to each row of uniform LBP's histogram because a cyclic shift of the LBP corresponds to a phase shift in its coefficients. This is a key point since invariance to rotation can be easily demonstrated [22].
Hence, LBP-HF achieves globally the rotation invariance because it works on the whole region through the histogram; moreover, this kind of features maintains information about the relative distribution of the orientation of the uniform local binary patterns.

4.4 Random Projection

Principal Component Analysis (PCA) allows the reduction of the dimensionality of a problem with best results in terms of mean-square error. Unfortunately, it has a really high computational cost which doesn't allow the application in the field of high-dimensional data sets. For this reason, a new method, called Random Projection (RP), has been implemented [24].
The reduction of dimensionality through RP is made by projecting the original high-dimensional data set h to a compressed space using a random matrix l whose columns have unitary size; the obtained matrix has h-by-l size. In this way, distances among points are approximately kept and, furthermore, a high-dimensional space has more almost-orthogonal than orthogonal directions: this means that randomly directed vectors might be approximately orthogonal. Vectors' similarity can then be calculated by Euclidean distance or inner product or any other possible method.
Comparing RP to other dimensionality reduction methods (like PCA) in noiseless images, noisy images and text data [24], Random Projection method shows a good computational efficiency and accuracy, since also Principal Component Analysis tends to distort data.

4.5 Filter banks

Here some filter banks are described [25]. They can be used in the field of texture classification. The Leung-Malik Filter Bank (LM) is a 48-filters bank: first and second derivatives of Gaussians at six orientations and three scales, 8 Laplacian of Gaussian's filters and 4 Gaussians. It is not invariant to rotation.
Instead, the Schmid Filter Bank (S) is made up by 13 rotationally invariant filters, which are of a circular shape and so isotropic.
Finally, the Root Filter Set (RFS) consists of 38 filters: a Gaussian, a Laplacian of Gaussian, an edge filter at three scales, a bar filter at the same three scales and two filters at six different orientations at each scale. It's then similar to the Leung-Malik filter bank.

Chapter 5

Frequency-based method

The frequency-based method used in this work is explained in [27]. The key point is the invariance to rotations since the searched patterns are orientated in different directions.
The original application of [27] was the automatic detection of sand ripple patterns in videos linked to underwater analysis. As mentioned in the Introduction, this way was chosen with the awareness of worse results than the methods used in [29],[28].
To deeply understand the steps followed in the implementation, some mathematical tools could be really useful. Hence, in Chapter 6 the results and their comments are reported.

5.1 Basic concepts and techniques

An image is a function of two continuous variables that has to be converted in a digital image by sampling and quantization in order to be processed. In this way, a digital image is a matrix of the same size of the original image, assumed M-by-N, in which each coordinate, called *pixel*, stands for a specific grey level in the range [0, L-1], where L is usually a power of 2 (binary encoding). Then, the histogram of the image is defined as a discrete function which shows how many pixels have the same grey level value, for each grey level value from 0 to L-1. A histogram normalization in [0,1] could be implemented by dividing each matrix component by

the total amount of pixels M·N; thus, the probability of any intensity occurrence is estimated.
Before any analysis, a pre-processing step could be advantageous to enhance the image. This step can be done in a huge variety of methods:

- An equalization of the histogram can be done to improve the contrast of the image defined as $C = \frac{\overline{I_a} - \overline{I_s}}{\overline{I_s}}$, where $\overline{I_a}$ is the mean value of the intensity of a region of interest and $\overline{I_s}$ is the mean value of the background intensity. If a histogram illustrates the majority of grey levels in a narrow range, the contrast is considered low and the equalization is required for better results.

- Some easy intensity transformations allow the conversion of a pixel grey value r into a new intensity s through n-th root or logarithm for lighting otherwise through n-th power or exponential for darkening. Furthermore, an image stretching can be implemented to extend a grey level values range to the whole possible range.

- Other transformations include the use of at least two images, in order to apply set (e.g. union, intersection) as well as arithmetic (e.g. product, average) operations. This is aimed at removing noise, changing brightness or produce masks.

- Spatial filters allow noise reduction both in a linear (e.g. moving average filter) and non-linear (e.g. median filter) way. Filters in frequency domain can improve the image contrast and the visualisation of details.

In this work, images were processed in the frequency domain. The bidimensional Fourier transform *F(u,v)* of a real function *f(x,y)* is stated as:

$$F(u,v) = \mathfrak{F}[f(x,y)] = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x,y) e^{-j2\pi(ux+vy)} dxdy. \tag{5.1}$$

Generally, the Fourier Transform is a complex function, hard to visualise, expressed as the sum of a Real part and an Imaginary part (j is the imaginary unit):

$$F(u,v) = R(u,v) + j \cdot I(u,v) \tag{5.2}$$

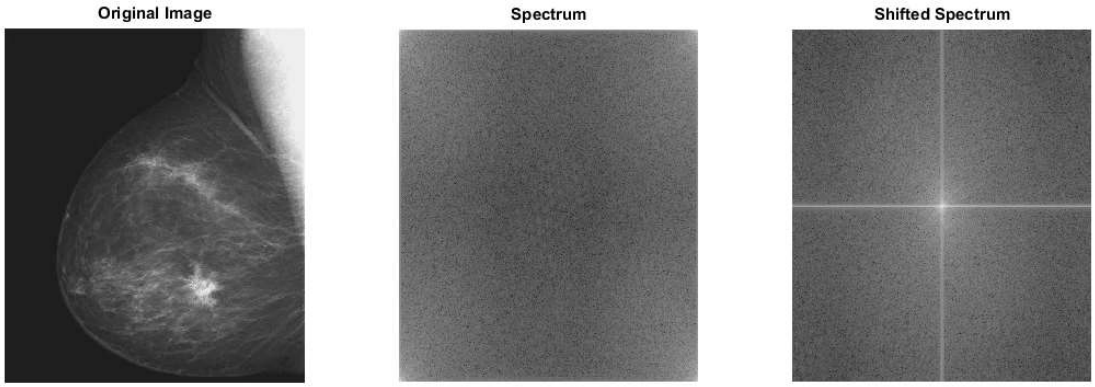


Figure 5.1: An example of shifting the 2D-FT in the centre of an image.

which can be written in terms of polar coordinates:

$$F(u,v) = |F(u,v)| \cdot e^{j \cdot \varphi(u,v)} \tag{5.3}$$

$$\text{where} \quad |F(u,v)| = \sqrt{R(u,v)^2 + I(u,v)^2} \quad \text{and} \quad \varphi(u,v) = \tan^{-1} \frac{I(u,v)}{R(u,v)}.$$

$|F(u,v)|$ and $\varphi(u,v)$ are called *spectrum* and *phase* of the transform, respectively. As the 1D-Fourier Transform, the 2D-FT has important properties, as [4] explains in Chapter 4; among all, there is the translation in the coordinates (x_0, y_0):

$$f(x - x_0, y - y_0) \Leftrightarrow F(u,v) \cdot e^{-j2\pi(x_0 u/M + y_0 v/N)} \tag{5.4}$$

The property in (5.4) is extremely important since it's necessary shifting the Fourier Transform of an image in its centre. As Figure 5.1 shows, the absolute value of the Fourier Transform needs to be shifted at the centre to have a better visualization (actually, there is also a log-transformation to increase details, otherwise the two images in the frequency field can not be adequately observed).

5.2 Method's theory

Repetitive patterns can be found by means of peaks in frequency, whose localization depends on ripples' orientation and spatial frequency. Thus, the best way to yield good results is deleting the dependence previously cited, achieving the rotation invariance, since the images of interest can show differently oriented stripes.
To achieve this goal, it's possible calculating the integral of the Fourier transform magnitude M (assumed square for simplicity) of the image in a circle of radius r centred in the origin. Mathematically, the computation to be done is:

$$d(r) = \int_{C_r} M(P) d\theta \tag{5.5}$$

in which θ is the angle blown away by a point P, moving along the circumference. In order to make d(r) always non-negative, it should be useful translating the coordinate frame in the centre of the region taken into consideration. As a consequence, the vector of the possible radii should have a value greater than 1 as minimum (in my case, I set 3) and a value less than half side of the square-assumed magnitude as maximum in order to remain into the chosen region.
Thus, *d(r)* has a maximum in the radius r that stands for the distance between the origin and the symmetric peaks. If a rotation of the image occurs, peaks rotate on the frequency plane, leading to a not significant variance in the position of d's maximum.
This method is implemented by using MatLab R2014b software. Hence, for each radius r, the function d(r) is calculated as the sum of the magnitudes for all angle values between 1° and 360° by step 1°.
Ideally, a pattern can be considered as a ripple pattern if it contains a relevant peak in the frequency domain. Before applying this technique to the real retinal images, the correct application of this method in the synthetic ones had to be proved. For this cause, images were built in a sinusoidal trend, i.e. as an alternation of white and black stripes at different frequencies, calculated in terms of cycles for image, that is the number of complete variations in that image. After the collection of the results, the application of the method for real images was done.
Of course, synthetic images bring different results from the real ones; this can be explained by the fact that synthetic images are binary so the stripes are really

clear if compared to the grey level values patterns. Furthermore, when visible, RNFL belongs to the high frequencies but, if studied in a window of size 50x50, can show different frequencies and orientations: this can compromise results too. In real images, the analysis of the peak located at radius r = 3 (the minimum set) comes from these observations since for greater radii there are no relevant peaks whereas for synthetic images there are also lower peaks than the one at r = 3.

Chapter 6

Experimental results

The first part of this Chapter regards the results obtained using synthetic images. The study was developed into three main stages:

1. variation of the function (5.5) due to frequency changes;
2. application of rotations to the images;
3. noise addiction.

After the collection of the results, the examination of the 884 retinal images follows in other three steps:

1. preprocessing for image enhancement;
2. setting of a threshold for the distinction between visible and invisible RNFL;
3. building of maps for images containing stripes.

6.1 Tests with synthetic images

When the frequency lowers, the peak of the function d(r) defined in (5.5) moved towards the origin, whereas it moved far from the origin if the frequency increased. It's demonstrated by Figure 6.1, where at a frequency of 50 cycles/image, the "M-shaped" peak began at r = 42, that was approximately twice the peak radius at half frequency. Analysing other frequencies, the position of the peaks changed according to the frequency: if the cycles for image were multiplied by a factor

k, the position of the peaks was shifted by a factor k too and vice versa. For this reason, the lower peaks achieved for radii greater than 60 for the case of 25 cycles/image were not present at the double frequency.

Furthermore, as shown in Figure 6.2a, rotating the image did not change too

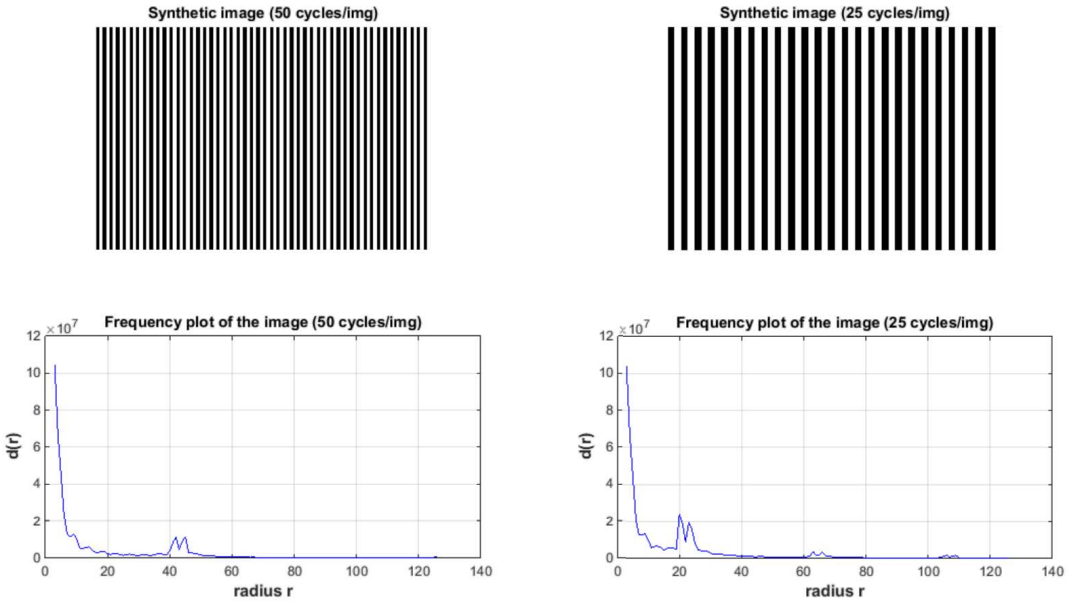
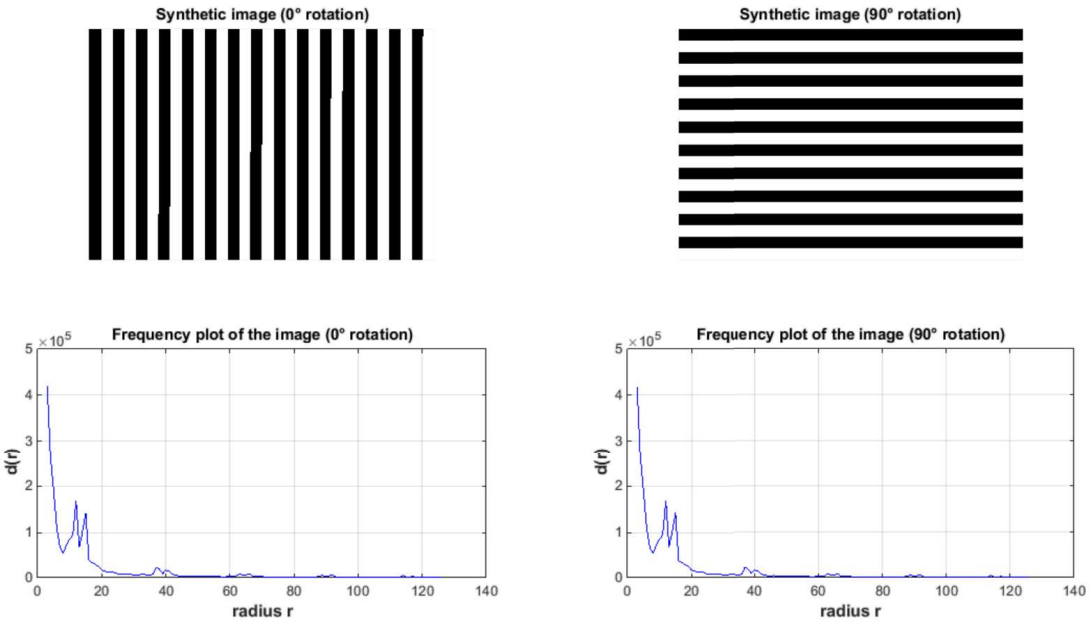


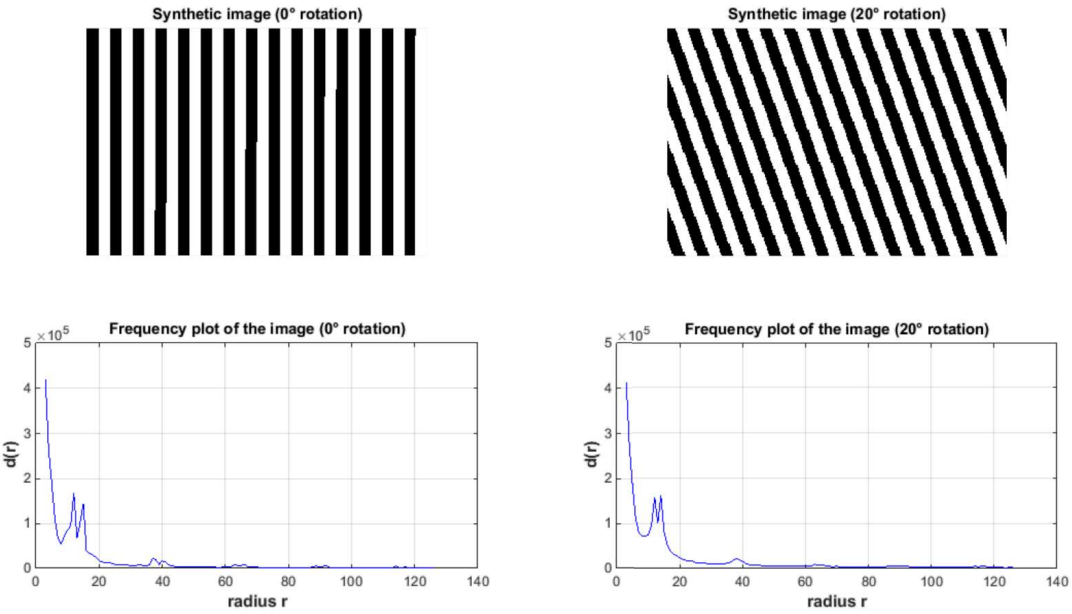
Figure 6.1: Changes in d(r) plot due to frequency's changes.

much the position of the peaks as literature states [27]. Actually, when the image was rotated by a number of degrees different from 0-90-180-270, there was a little alteration in peak's position and amplitude. This fact became clear looking at MatLab function `imrotate` used for this action: the original image was rotated in such a way it was completely contained in the new image, leading to a larger image characterized by black angles. To reduce these effects, the original image was firstly enlarged, then rotated and ultimately cut, so that a window of the desired dimensions could be taken inside it, without taking into account not interesting components. Nevertheless the rotated images showed a quite different trend both among them and from the original image and the ones rotated by 90°-180°-270°: the position of the peak kept being localised at the same radius but its amplitude and shape may vary. In fact d(r) showed small variations in its trend if compared to 0°'s, 90°'s or their multiples' trend, as can be seen in Figure 6.1b; remarkable differences could be noticed for rotation angles in the approximate gap [30°÷65°] and the correspondent intervals in the second, third and fourth quarters, where

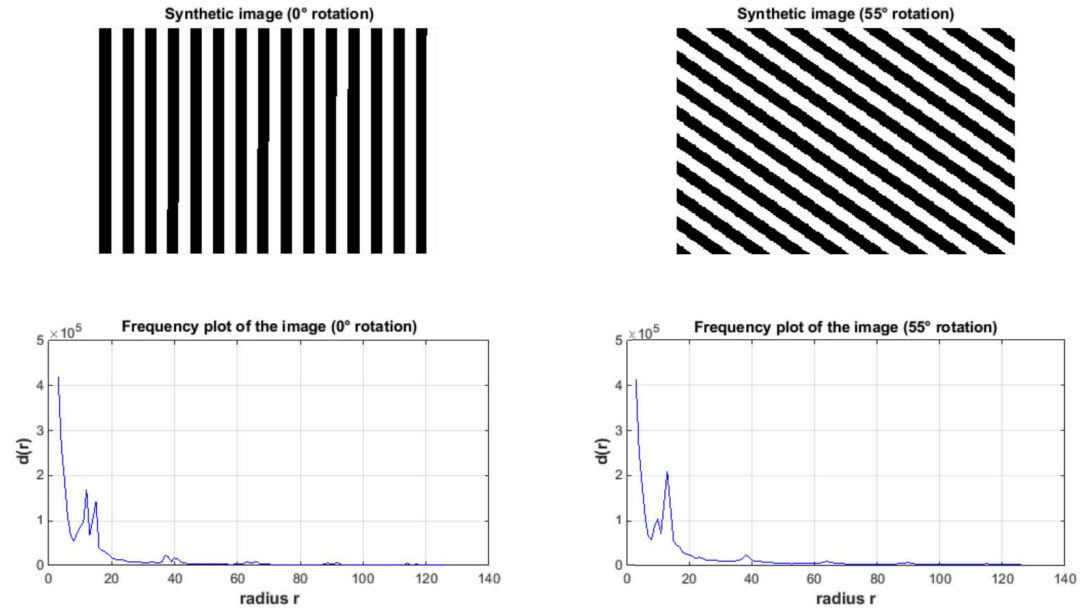
the shape of the peaks changed almost completely (Figure 6.1c).
Some considerations must be done. Rotated images were set to have the same size of the 0° rotated image because of the already mentioned problem linked to the MatLab function. This meant that their contents were turned, but the images themselves were not: synthetic images, illustrated in Figure 6.1b, as well as in Figure 6.1c, were not sloping by 20° or 55°, respectively, compared to the horizontal axis but only their stripes were. As a consequence, these images had a different frequency than the original one, in terms of cycles/image, and this could be why trends were altered.
To move close to real images, synthetic images were then corrupted by *salt and pepper* noise, which randomly adds white and black pixels.
As Figure 6.2 shows, noise brought to different results: for small densities, the frequency plot didn't change too much if compared to the original image one, whereas the more relevant the noise was, the less significant the peaks of d(r) were. In fact, for a noise density of 0.6 or more, the image was hardly recognizable and the stripy pattern was more imprecise than the previous case.



(a) Frequency plot for 0° and 90° rotation.



(b) Frequency plot for 0° and 20° rotation.



(c) Frequency plot for 0° and 55° rotation.

Figure 6.1: Frequency plots for a synthetic image (15 cycles/image) and its rotations.

This fact implicates that for real images a well defined trend for the function in (5.5) can not be achieved since retinal images are affected by noise due to the acquisition and the devices used to data capture; moreover, RNFL are not equal to the ideal texture analysed so far.
Nevertheless, all these preliminary studies were considered a good achievement for the evaluation of the method in the case of synthetic images. Thus, it was possible to evaluate retinal images.

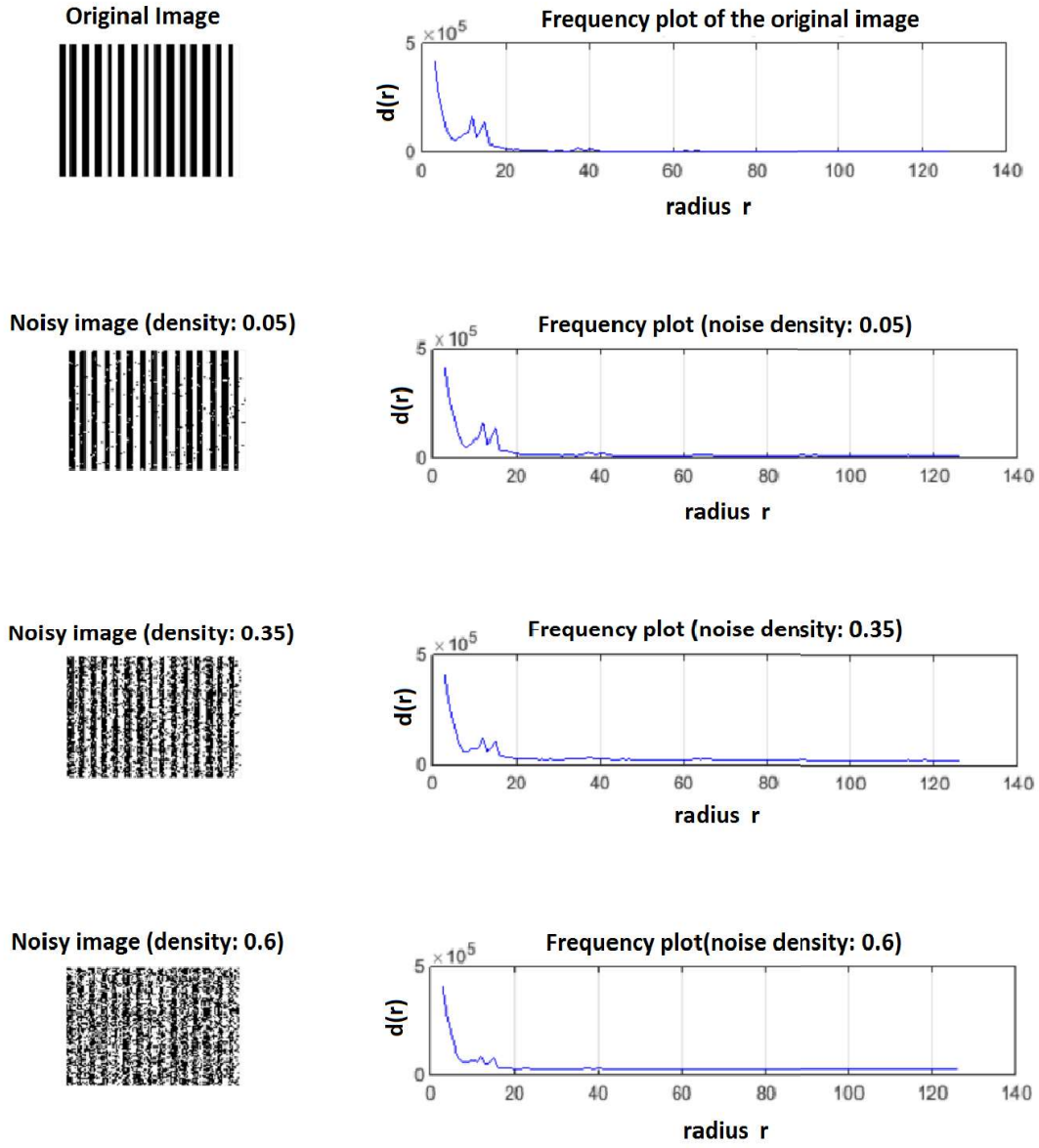


Figure 6.2: Synthetic image corrupted by *salt and pepper* noise at increasing densities.

6.2 Tests with real retinal images

A pre-processing step was necessary since real images are characterised by a quite low contrast and are sometimes noisy too. The path followed at this stage was made up by the subsequent actions:

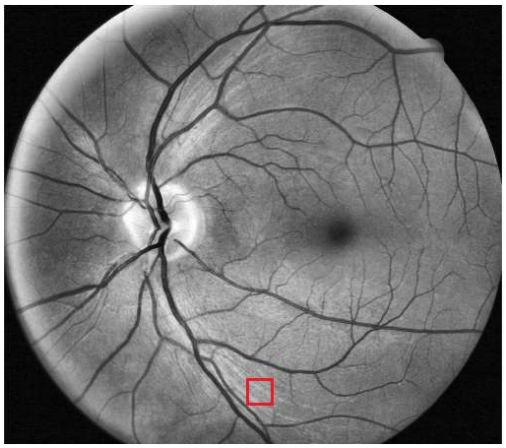
- Image Conversion: each RGB image is considered only on the Green channel because of its higher contrast. In this way, every image is converted into a grey-level image;
- Normalization: since the values achieved for d(r) (Equation (5.5)) are really high for real images, the normalization allows limited results;
- Equalization: at this point an enhancement of the contrast can be done in order to make more visible ripple patterns.

First of all, a few images among the 884 available images were analysed. It was necessary splitting the images in windows because the repetitive patterns are not over the whole image, but only in specific regions of it, as Figure 1.1 reported. These windows were set to be squared and to have size $50x50$ pixels, since larger boxes could hardly contain only ripple patterns, without blood vessels.
The detection had to be done in a manual way as a first step. Ripple patterns in fundus camera images are hard to find since they can be *visible* or *invisible*. To understand the frequency behaviour, the problem was initialised searching for windows belonging to three different classes:

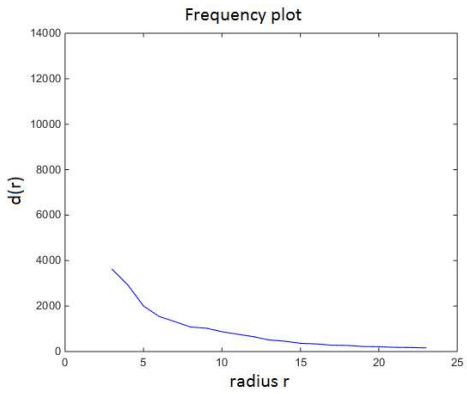
- a first class for only retinal nerve fiber layer;
- a second class for both stripes and blood vessels;
- a third class for the windows containing neither RNFL nor blood vessels.

These windows were taken from different images of the original dataset and the algorithm previously described was applied to them.

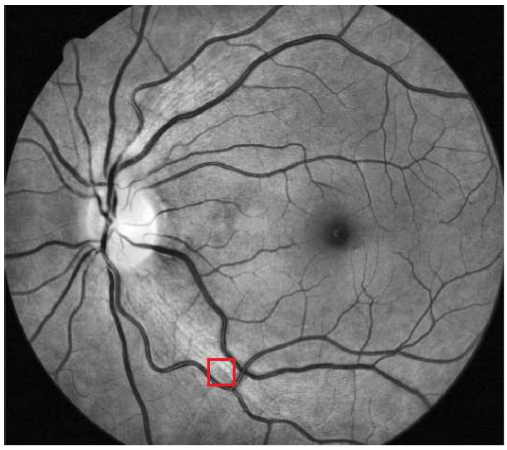
As Figure 6.2e and Figure 6.2f show, windows containing nothing, i.e. neither stripes nor blood vessels, were characterised by a frequency plot of d(r) which had a really low peak, that was high 824.7 at radius $r = 3$ in the specific reported example. This trend was similar for all the windows of this class.



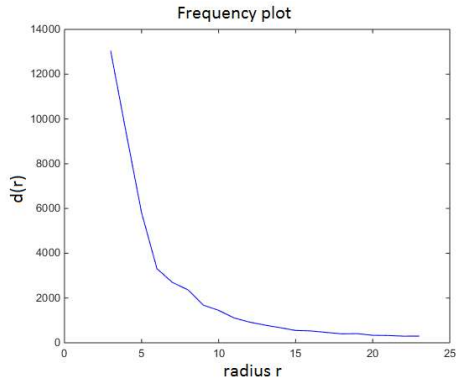
(a) An example for the first class.



(b) The frequency plot for first class' example.

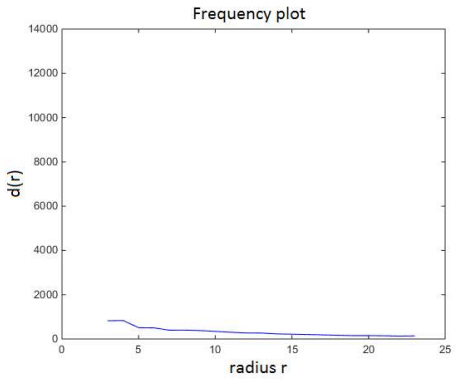


(c) An example for the second class.



(d) The frequency plot for the second class' example.

(e) An example for the third class.



(f) The frequency plot for the third class' example.

Figure 6.2: Some examples for the defined classes. (a)-(b) for windows containing RNFL, (c)-(d) for windows containing also blood vessels, (e)-(f) for empty windows.

In the two remaining examples of Figure 6.2, stripes were present. If a window included only RNFL, the peak was higher than the previous case (it achieved 3616) but lower than the one reached by the mixing case. In Figure 6.3d the highest peak was noticeable and, studying more images, the more capillaries, veins and arteries were visible, the higher the peak was; hence, stripy patterns were significantly influenced by blood vessels if studied in the frequency domain. This could be explained by the fact that vessels had a better level of contrast and were thicker than RNFL; thus, stripy patterns showed a lacking visibility.

However, windows characterized by RNFL showed to be extremely different from the ideal pattern in terms of curve's shape. Reasons could be numerous: images were taken in grey-level instead of binary levels; each window could contain a non-uniform frequency pattern which could bring to a modification of the frequency behaviour; noise, even if reduced at the pre-processing step, could affect the calculation of d(r) entailing a maximization of the situation depicted in Figure 6.2, for the case of a synthetic image corruption by a *salt and pepper* noise by density of 0.6.

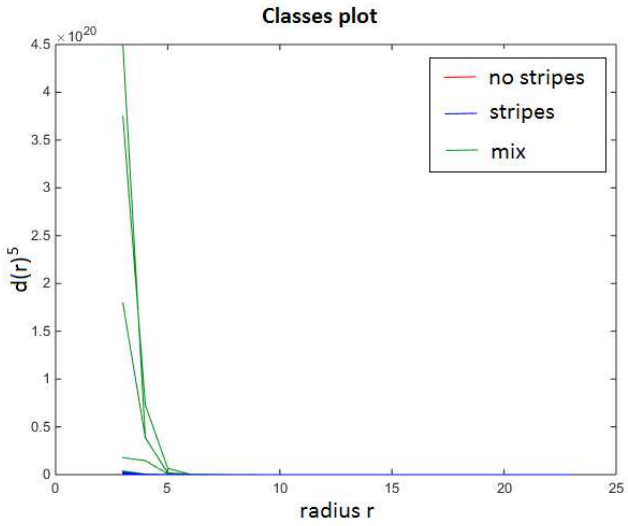
In order to find a threshold that allowed the classification of a window as characterized by retinal nerve fiber layer or not, d(r) trends belonging to different classes were held in the same plot, choosing 10 elements for the first class, 7 for the second one and 6 for the third one. The small number of objects for each class was due

to two main reasons. One cause was that plots were similar in the same class thus choosing a limited number of windows for each group permitted a better visualization of the peculiar trends. The second cause was the complexity of finding several windows of interest; for instance, $50x50$ boxes characterized only by ripple patterns were really hard to find since they were localized along the directions of blood vessels. Furthermore, in some images capillaries were so thin that could be confused with one of the lines of the alternating pattern.
The function d(r) of the common frequency plot was elevated to the 5th power in order to stress the differences among classes (Figure 6.3). The height of the peaks in case of blood vessels' presence was so elevated that it flattened the other two classes to very low frequencies; actually, as Figure 6.3b shows, when doing a zoom between radii $r = 3$ and $r = 3.3$, classes remain quite distinct except for some outliers that could come from classification errors of a window.

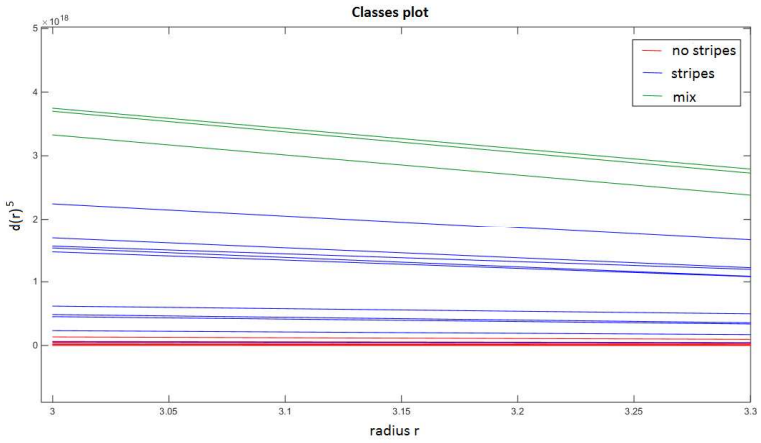
Two main problems were found:

- The difference between the class of only stripes (blue in Figure 6.3b) and the class with nothing (red) is noticeable. Nevertheless, the red class lay below $3 \cdot 10^{17}$ which, even if it was a quite good threshold for the classification "stripes" or "not stripes", it was also a really low value that cannot allow an optimal classification of the image since most windows have peaks from $4 \cdot 10^{17}$ to $4 \cdot 10^{20}$ or more. On one hand, this supported the fact that RNFL and noise could be confused in the frequency domain; on the other hand, when studying all the windows in an image, it could be likely having an extremely high number of windows defined with stripes. Hence, in both cases the classification problem could be tough to solve.

- This method could do edge detection too. In fact, it recognized black angles outside the eye as background and this led to classify all the windows containing both background and image in the "RNFL" class, even if there was a single blood vessel. This was supported by the study of a synthetic image of frequency of 1 cycle/image, whose d(r) plot was predominantly flat if compared to the images in Figure 6.1, situation that simulated the presence of a background behind the object of interest.



(a)



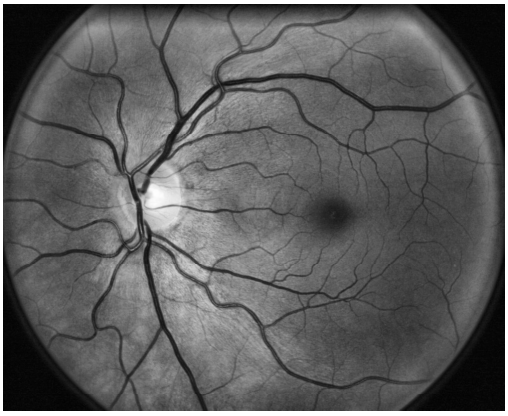
(b)

Figure 6.3: The common frequency plot for the three classes (a) and its zoom (b).

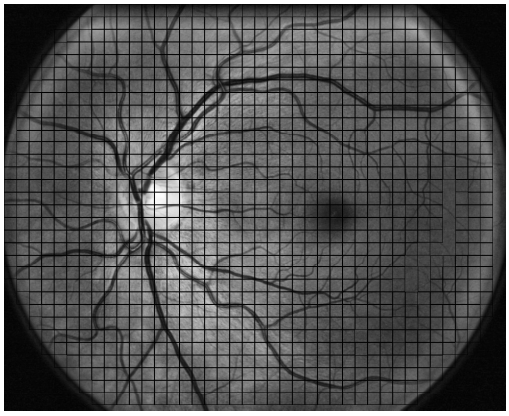
From these observations and some tries, the threshold was set to $T = 6 \cdot 10^{15}$. Low values tended to classify the whole image as containing RNFL, too much higher values tended instead to ignore windows that included only stripy patterns.
Now the analysis of the whole dataset could be done. The procedure previously described was implemented for all images: at first, pre-processing of the global image, hence the setting of window's size (50-by-50) and at the end the application of the technique locally. For each image, a window was moving through the rows and the columns of the image, with step of 25 pixels: this meant that, watching the totality of the boxes in the retinal image, a window overlapped with the adjacent ones both in the horizontal and in the vertical directions in order to analyse a greater number of regions. A second threshold was set to avoid the classification of those windows that contained background: if the majority of the pixels of a box was dark, the method wasn't applied and the attempt was done to the next window. If the frequency based procedure was used in a window, the function (5.5), elevated to the 5th power, was compared to the threshold T: if it was greater than T, the count of the RNFL patterns was increased by 1, otherwise nothing happened.
Simultaneously, a map of the regions classified as containing ripple patterns was built; this was done in a binary domain, considering the studied window as a white box in a black image I: the boundary extraction was done through I - *erosion(I,s)*, if I is the current image and s is a squared structural element[1]. The subtraction enabled the detection of the edges of the window of interest to draw in the original image. Hence, after the probe of the whole image, each map was drawn all over the corresponding image (Figure 6.4).

As expected, the method didn't work well for RNFL detection. As Figure 6.4 shows, maps appeared in the case of both visible and invisible RNFL; furthermore, stripy patterns were detected not only along the directions that went towards the optic nerve but also in the proximity of the fovea. This could be probably due to the wrong set threshold even if results didn't improve by trying other thresholds. On average, indeed, 85.9% of image's windows were classified as containing RNFL.

[1]The *erosion* is a morphological operator which acts in the following way: if A,B belong to $\mathbb{Z}^2$, *erosion* $= \{z|(B)_z \subseteq A\}$, where A is the binary image, B is an adequate structuring element and $(B)_z$ is the reflection of B. A structuring element is a subset or a sub-image which helps the exploration of an image in order to find regions of interest.

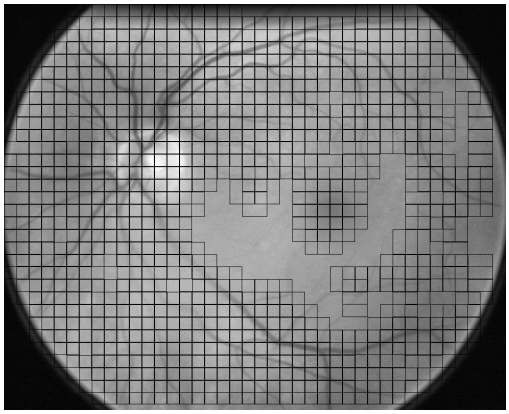


(a) Preprocessed image.



(b) Map of the image in (a).

(c) Preprocessed image.



(d) Map of the image in (c).

Figure 6.4: Maps resulting from the applied method: (a)-(b) for visible RNFL, (c)-(d) for invisible RNFL.

Chapter 7

Conclusions

The aim of this thesis was to find a frequency based method that could detect Retinal Nerve Fiber Layer in retinal images, comparing its efficiency to the techniques used at School of Computing, University of Dundee, UK.
To better understand how feature extraction used in [28],[29] worked, literature about such techniques was studied. After the study of papers concerning methods able to detect stripy patterns, the procedure described in [27] was implemented for the application of this thesis' interest.
Thus, the function explained in Equation (5.5) wanted to calculate the integral of the Magnitude of the Fourier Transform along a circumference to achieve rotation invariance since the images that had to be analysed in this thesis presented stripy patterns differently oriented. Hence, in a first stage, synthetic images were built to study the properties and the behaviour in frequency of the ideal case; this was followed by the analysis of retinal images.
In the ideal case, results were consistent with literature studies, since changing frequencies led to variations in the function d(r); limited modifications in the frequency behaviour were justified by the fact that MatLab has some bounds when it has to rotate an image. An interesting further study could be the research of a software function which allows the actual rotation of an image.
In the real case, it was clear that the frequency based method implemented in the thesis wasn't appropriate to this application. Of course, one of the problems could be the setting of the threshold but, since the visibility of the RNFL is really low, such that RNFL could almost be considered as noise, the investigation of an-

other threshold seemed pretty tough. Furthermore, the high contrast of the blood vessels made harder the detection of the RNFL: the contrast of the whole image, even if enhanced in a pre-processing step, didn't allow a more visible view of the patterns looked for in this thesis. All the mentioned negative results were spread to the whole dataset: this meant that the 884 retinal images were all classified as containing RNFL.

The experiment led in [29] with a different dataset but the same aim brought to an improvement of the performance over the state-of-the-art: the SIFT method gave an higher agreement with the most experienced clinicians (over the 80%). The frequency based method, actually, agreed with the annotators every time there was an occurrence of an image with visible RNFL, otherwise it disagreed: this stressed the failure of this method for the interested goal.

An improvement can be done taking into consideration some *a priori* information about the probability of having a ripple pattern in a certain region (along the vessels that go towards the optic nerve) rather than in others (in the proximity of the fovea). Hence, calculating d(r) with this knowledge could lead to better results.

In the end, some other methods in the frequency domain could be tried. For instance, the use of the Wavelet Transform could be helping if implemented as in the case of removing noise thorough the image but in a reverse way: instead of deleting noisy components (among which there will probably be RNFLs) with Wavelets, the good part of the image is removed and thus only noise and stripy patterns are kept.

Bibliography

[1] K. L. Thomson, J. M. Yeo, B. Waddell, J. R. Cameron, S. Pal, *A systematic review and meta-analysis of retinal nerve fiber layer change in dementia, using optical coherence tomography*, Alzheimer's & Dementia: Diagnosis, Assessment & Disease Monitoring 1 (2015), pp. 136-143

[2] *What is dementia?*, available at: `https://www.alzheimers.org.uk/info/20007/types_of_dementia/1/what_is_dementia`, consultation date: 30/01/2017

[3] M. P. Saccomanni, *Corso LM di Bioimmagini*, Università degli Studi di Padova, Academic Year 2015/2016

[4] R. C. Gonzales, R. E. Woods, *Digital Image Processing*, Third Edition, Prentice-Hall, 2008

[5] M. S. Nixon, A. S. Aguado, *Feature Extraction & Image Processing for Computer Vision*, Third Edition, Elsevier, 2012

[6] Human Eye, available at: `https://www.britannica.com/science/human-eye/images-videos`, consultation date: 07/01/2017

[7] H. Kolb, *Simple Anatomy of the Retina*, available at: `http://webvision.med.utah.edu/book/part-i-foundations/simple-anatomy-of-the-retina/`, consultation date: 07/01/2017

[8] M. D. Abramoff, M. K. Garvin, M. Sonka, *Retinal Imaging and Image Analysis*, IEEE Transactions on Medical Imaging, 2010

[9] Y. Cai, G. Baciu, *Detection of Repetitive Patterns in Near Regular Texture Images*, IVMSP Workshop, IEEE, 2011, pp.60-65

[10] B. Julesz, *Textons, the elements of texture perception, and their interactions*, Nature, vol.290, n. 5802, pp. 91-97, 1981

[11] Y. Cai, G. Baciu, *Detecting, Grouping, and Structure Inference for Invariance Repetitive Patterns in Images*, IEEE Transactions on Image Processing, vol. 22, n. 6, pp. 2343-2355, June, 2013

[12] R. Wang, W. Li, C. Li, *A Method of Stripe Feature Extraction for Scrap Paper Image Mosaics*, 2013 Ninth International Conference on Natural Computation (ICNC), IEEE, pp. 1214-1218, 2013

[13] P. Sobrevilla, E. Montseny, M. Grau-Sánchez, *A New Fuzzy-Based System for Extracting Structural Features allowing to Capture Stripes' Image Local Patterns*, System Conference, 2009 3rd Annual IEEE

[14] D. He, L. Wang, *Texture Unit, Texture Spectrum, and Texture Analysis*, IEEE Transaction On Geoscience and Remote Sensing, Vol. 28, No. 4, July 1990

[15] J. Hayes, *-Image Classification- Gray Level Co-Occurrence Matrix (GLCM)*, available at: `http://web.pdx.edu/~jduh/courses/Archive/geog481w07/Students/Hayes_GreyScaleCoOccurrenceMatrix.pdf`, consultation date: 13/03/2016

[16] A. Nasri, R. Benslimane, *Symmetrical motif extraction for Periodic Ornamental Patterns*, International Conference on Multimedia Computing and Systems (ICMCS), IEEE, 2014

[17] C. Schmid, *Bag-of-features for category classification*, available at: `http://www.di.ens.fr/willow/events/cvml2011/materials/CVML2011_Cordelia_bof.pdf`, Consultation date: 09/12/2016

[18] M. Tappen, *Recognition with Bag-of-Words*, available at: `http://www.cs.ucf.edu/~mtappen/cap5415/lecs/lec17.pdf`, consultation date: 13/03/2016

[19] D. G. Lowe, *Object Recognition from Local Scale-Invariant Features*, Proc. of the International Conference on Computer Vision, Corfu, 1999

[20] E.H. Adelson, C.H. Anderson, J.R. Bergen, P.J. Burt, J.M. Ogden, *Pyramid methods in image processing*, RCA Engineer, 29-6, Nov/Dec 1984

[21] M. Pietikainen, J. Heikkila, *Tutorial: Image and Video Description with Local Binary Pattern Variants*, University of Oulu, CVPR 2011

[22] T. Ahonen, J. Matas, C. He, M. Pietikainen, *Rotation Invariant Image Description with Local Binary Pattern Histogram Fourier Features*, Image Analysis, SCIA 2009, LNCS 5575, pp. 61-70, 2009

[23] T. Jabid, Md. H. Kabir, O. Chae, *Local Directional Pattern (LDP) for Face Recognition*, Digest of Technical Papers International Conference on Consumer Electronics, IEEE, 2010

[24] E. Bingham, H. Mannila, *Random Projection in dimensionality reduction: Application to image and text data*, ACM SIGKDD International Conference on Knowledge discovery and data mining, New York, USA, pp. 245-250, 2001

[25] M. Varma, A. Zisserman, J. Geusebroek, *Texture Classification*, available at: `www.robots.ox.ac.uk/~vgg/research/texclass/filters.html`, Visual Geometry Group, Department of Engineering Science, University of Oxford, consultation date: 20/04/2016

[26] S. Mavandadi, P. Aarabi, K. N. Plataniotis, *Fourier-Based Rotation Invariant Image Features*, International Conference on Image Processing (ICIP), 16th IEEE, 2009

[27] A. Olmos, M. Trucco, K. Lebart, D. Lane, *Detecting ripple patterns in Mission Videos*, OCEANS 2000 MTS/IEEE Conference and Exhibition, vol. 1, pp. 331-336, 2000

[28] S. Manivannan, W. Li, S. Akbar, R. Wang, J. Zhang, S. J. McKenna, *An automated pattern recognition system for classifying indirect immunofluorescence images of HEp-2 cells and specimens*, Pattern Recognition, 51, pp. 12-26, Elsevier, 2015

[29] S. Manivannan, C. Cobb, S. Burgess, E. Trucco, *Sub-Category Classifiers for Multi-Instance Learning and its application to Retinal Nerve Fiber Layer*

Visibility Classification, Medical Image Computing and Computer-Assisted Intervention (MICCAI), MICCAI, 2016