University of Northumbria
Department of Computing and Information Sciences

Automated Classification of Cancer Tissues using Multispectral Imagery

Rémy Peyret

A thesis submitted in part fulfilment of the requirements for the degree of Doctor of Philosophy in Computing of the University of Northumbria, October 2017

Declaration

I declare that the work contained in this thesis has not been submitted for any other award and that it is all my own work. I also confirm that this work fully acknowledges opinions, ideas and contributions from the work of others.

Any ethical clearance for the research presented in this thesis has been approved. Approval has been sought and granted by the Faculty Ethics Committee on 04/01/2016.

I declare that the Word Count of this Thesis is 40861 words

Rémy Peyret

October 2017

À Jacky.

'Nobody phrases it this way, but I think that artificial intelligence is almost a humanities discipline. It's really an attempt to understand human intelligence and human cognition.'

Sebastian Thrun

Acknowledgements

Before moving on to the persons who helped to shape my PhD experience, I would like to acknowledge the full financial support provided by the Qatar Research Foundation without which this project would not have existed.

First, I would like to express my utmost gratitude to my supervisor Pr. Ahmed Bouridane for trusting me and giving me the opportunity of conducting this PhD study. His continuous guidance, support and advice have been a considerable help before I even started my PhD and throughout the phases of research and writing of this thesis.

I would also like to thank my second supervisor Dr. Fouad Khelifi for his judicious advice and rigorous feedback at every step of this research project. His technical input was always constructive and pertinent, and I was guided by our enlightening discussions.

My gratitude extends to Dr. Muhammad Atif Tahir for his collaboration to this work and his knowledge of histopathological data and machine learning.

My sincere thanks go to the Qatar team and more specifically to Pr. Somaya Al-Maadeed and Suchithra Kunhoth for welcoming me in their country and making me take part of the image acquisition process. Thanks to Dr. Rafif Al-Saady for her medical counsel on histology and colorectal cancer recognition. Her help was critical for understanding the work of pathologists.

An immense thank you to my team of proofreaders: Guillaume Raffaud, Rayana Boubezari, Arezoo Amirkhalili, Sebastian Prost and Bridie Chomse. Your rigour and precision made reading this thesis less of an unpleasant chore for everyone.

To Rayana, thank you for encouraging me to accept this PhD offer and for being such a close friend and colleague of mine throughout these three years.

Those lunch breaks livened up by our passionate discussions, our gym sessions and your moral support undoubtedly made my PhD substantially more enjoyable. Thanks to Arezoo for bearing with us when we were speaking French.

I would like to offer special thanks to Kahina, David, and Pazza who made the lab F7 a more social place in the last year of my PhD. Good luck in your new building!

To my housemates, Felix, Megan, Seb, Simon, Till, and Tom, a gigantic thank you for making the house such a pleasant place to live in. Thank you for your moral support, for the shared dinners, BBQs, parties and pink knickers. A particular thanks to Felix and Megan for accommodating me in the last few months of my PhD and bearing with my stress during my writing stage. These thanks have to be extended to all of my Newcastle friends who have always been supportive and have helped me balance my work and social life. I owe you guys not loosing my mind.

To my friends from "Centrale Marseille", I am grateful for those amazing trips to the other side of the world. They definitely were part of my PhD experience and hopefully will continue for the years to come. Spéciale dédicace à la team des thésards (crocodile, c'est placé !) et aux Mégères.

To my parents, who do not speak english and my sister, Mathilde: merci d'être une famille si aimante et attentionnée. Vous m'avez permis d'être qui je suis. À Olivier, merci de tes conseils qui m'ont guidé tout au long de mes années d'études. Merci aux cousins grenoblois de m'avoir si bien accueilli à mon départ de la Guadeloupe, je n'aurais certainement pas réussi mes études de la même façon sans vous. À mes grands-parents, à ma marraine, Régine, merci de votre amour bienveillant.

Abstract

Automated classification of medical images for colorectal and prostate cancer diagnosis is a crucial tool for improving routine diagnosis decisions. Therefore, in the last few decades, there has been an increasing interest in refining and adapting machine learning algorithms to classify microscopic images of tumour biopsies. Recently, multispectral imagery has received a significant interest from the research community due to the fast-growing development of high-performance computers. This thesis investigates novel algorithms for automatic classification of colorectal and prostate cancer using multispectral imagery in order to propose a system outperforming the state-of-the-art techniques in the field.

To achieve this objective, several feature extraction methods based on image texture have been investigated, analysed and evaluated. A novel texture feature for multispectral images is also constructed as an adaptation of the local binary pattern texture feature to multispectral images by expanding the pixels neighbourhood to the spectral dimension. It has the advantage of capturing the multispectral information with a limited feature vector size. This feature has demonstrated improved classification results when compared against traditional texture features. In order to further enhance the systems performance, advanced classification schemes such as bag-of-features – to better capture local information – and stacked generalisation – to select the most discriminative texture features – are explored and evaluated. Finally, the recent years have seen an accelerated and exponential rise of deep learning, boosted by the advances in hardware, and more specifically graphics processing units. Such models have demonstrated excellent results for supervised learning in multiple applications. This observation has motivated the employment in this thesis of deep neural network architectures, namely convolutional neural networks. Experiments were also carried out to evaluate and compare the performance obtained with the features extracted using convolutional neural networks with

random initialisation against features extracted with pre-trained models on ImageNet dataset. The analysis of the classification accuracy achieved with deep learning models reveals that the latter outperforms the previously proposed texture extraction methods. In this thesis, the algorithms are assessed using two separate multiclass datasets: the first one consists of prostate tumour multispectral images, and the second contains multispectral images of colorectal tumours. The colorectal dataset was acquired on a wide domain of the light spectrum ranging from the visible to the infrared wavelengths. This dataset was used to demonstrate the improved results produced using infrared light as well as visible light.

List of Publications

Peer-Reviewed Publications:

- R. Peyret, A. Bouridane, F. Khelifi, M. A. Tahir, and S. Al-Maadeed, "Automatic classification of colorectal and prostatic histologic tumor images using multiscale multispectral local binary pattern texture features and stacked generalization," *Neurocomputing*, 2017
- R. Peyret, A. Bouridane, S. A. Al-Maadeed, S. Kunhoth, and F. Khelifi, "Texture analysis for colorectal tumour biopsies using multispectral imagery," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, pp. 7218–7221
- R. Peyret, F. Khelifi, A. Bouridane, and S. Al-Maadeed, "Automatic Diagnosis of Prostate Cancer using Multispectral based Linear Binary Pattern Bagged Codebooks," in *2017 International Conference on Bioengineering for Smart Technologies (BioSMART 2017)*, Aug. 2017
- S. Al Maadeed, S. Kunhoth, A. Bouridane, and R. Peyret, "Multispectral imaging and machine learning for automated cancer diagnosis," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1740–1744, IEEE, June 2017

Publication Under Preparation:

- R. Peyret, A. Bouridane, F. Khelifi, S. Al-Maadeed, "Convolutional Neural Networks for Automatic Classification of Colorectal and Prostate Tumour Biopsies using Multispectral Imagery"

Table of Contents

List of Tables

List of Figures

Nomenclature

Acronyms

k-**NN**	**k**-**N**earest **N**eighbour
2D	**2**-**D**imension(al)
3D	**3**-**D**imension(al)
AUC	**A**rea **U**nder **C**urve
BoF	**B**ag-**o**f-**F**eatures
BoW	**B**ag-**o**f-**W**ords
BPH	**B**enign **P**rostatic **H**yperplasia
CADS	**C**omputer-**A**ided **D**iagnosis **S**ystsem
CAD	**C**omputer-**A**ided **D**iagnosis
Ca	**Ca**rcinoma
CCD	**C**harge-**C**oupled **D**evice
CNN	**C**onvolutional **N**eural **N**etwork
CPU	**C**entral **P**rocessing **U**nit
DT	**D**ecision **T**ree

FNR	**F**alse **N**egative **R**ate
FN	**F**alse **N**egative
FPR	**F**alse **P**ositive **R**ate
FP	**F**alse **P**ositive
GLCM	**G**rey-**L**evel **C**o-occurrence **M**atrix
GPU	**G**raphics **P**rocessing **U**nit
HP	**H**yperplastic **P**olyp
ICA	**I**ndependent **C**omponent **A**nalysis
IR	**I**nfra-**R**red
LBP-TOP	**L**ocal **B**inary **P**attern-**T**hree **O**rthogonal **P**lan
LBP	**L**ocal **B**inary **P**attern
LCTF	**L**iquid **C**rystal **T**unable **F**ilter
LDA	**L**inear **D**iscriminant **A**nalysis
LIOP	**L**ocal **I**ntensity **O**rder **P**attern
LR	**L**ogistic **R**egression
LTP	**L**ocal **T**ernary **P**attern
MLP	**M**ulti-**L**ayer **P**erceptron
MMLBP	**M**ultispectral **M**ultiscale **L**ocal **B**inary **P**attern
NRP	**N**o **R**emarkable **P**athology
OvsA	**O**ne-**v**ersu**s**-**A**ll
OvsO	**O**ne-**v**ersu**s**-**O**ne

PCA	**P**rincipal **C**omponent **A**nalysis
PCa	**P**rostatic **Ca**rcinoma
PIN	**P**rostatic **I**ntraepithelial **N**eoplasia
PPMM	**P**robabilistic **P**airwise **M**arkov **M**odel
pp	**p**ercentage **p**oint
ReLU	**Re**ctified **L**inear **U**nit
RF	**R**andom **F**orest
RGB	**R**ed **G**reen **B**lue
ROC	**R**eceiver **O**perating **C**haracteristic
SIFT	**S**cale-**I**nvariant **F**eature **T**ransform
Str	**Str**oma
SURF	**S**peeded **U**p **R**obust **F**eatures
SVM	**S**upport **V**ector **M**achine
TA	**T**ubular **A**denoma
TNR	**T**rue **N**egative **R**ate
TN	**T**rue **N**egative
TPR	**T**rue **P**ositive **R**ate
TP	**T**rue **P**ositive
Vis	**Vis**ible
VLBP	**V**olume **L**ocal **B**inary **P**attern
WEKA	**W**aikato **E**nvironment for **K**nowledge **A**nalysis

Symbols

$\mathbb{E}$	Mathematical expectation
$\nabla_\theta J(\theta)$	Gradiant of the cost function J along θ
$\mathbf{x}$	vector x
$\mathbf{x}^T$	Transposed vector x
$GLCM_{r,\theta}$	Co-occurence matrix of an image for the specific spatial relationship (r, θ), r being a real number representing the radius and θ a real for the angle.
J	Cost function
$LBP^{\lambda}_{P_\lambda,R}$	Local binary pattern for the spectral plans of a pixel for a set of P_λ neighbour spectral plans and a distance R
$LBP^{riu2}_{P,R}$	Rotation invariant uniform local binary pattern of a pixel for a set of P neighbours and a radius R
$LBP_{P,R}$	Local binary pattern of a pixel for a set of P neighbours and a radius R
$MMLBP_{P,P_\lambda,R}$	Multispectral Multiscale local binary pattern of a pixel for a set of P neighbours and P_λ neighbour spectral plans, and a radius R
U	Uniformity measure
$VLBP_{P,R}$	Volume local binary pattern of a pixel for a set of P neighbours and a radius R

Chapter 1

Introduction

1.1 Introduction

The World Health Organization has declared that the cancer burden is a worldwide health problem. According to their 2014 report, 14 million new cases were diagnosed in 2012 and 8 million people died from cancer in the same period [5]. Colorectal cancer is the third most common cancer globally and prostate is in second position amongst men representing respectively 9.7 % and 7.9 % of all cancers for both sexes [5]. Both colorectal and prostate tissues are glandular and therefore have a similar histological appearance. They are also both subject to the same tumor types; adenocarcinoma being the most commonly diagnosed cancerous tumor type in these organs. The known incidence of these cancers has been growing rapidly, partly due to increased life expectancy but also because of better public awareness of the diseases, which has led to higher performing and more frequent diagnosis tests [6]. For prostate cancer diagnosis, the European Association of Urology's guidelines [7] a histological analysis carried out on a sample taken from a needle biopsy. In a needle biopsy, a

small sample of tissue is removed from the prostate gland and prepared for microscope examination with precise staining and sliding procedures. The histological analysis is then performed by a highly trained pathologist, who uses a microscope to visually navigate over the biopsy sample slide. The pathologist finally decides the grade and stage of the cancer or the type of tumour based on their experience and expertise. This diagnosis is crucial for determining a course of treatment [8] and is also the most widely used method for colorectal cancer diagnosis [9]. However, this process is very laborious and time-consuming for pathologists, as they have to manually analyse every sample to spot the particular features characterising the type of tumor and the various cancer stages. It results in a high intra- and inter-observer variability [10, 11] which affects the reliability of the diagnosis. In december 1999, a study [12] of more than 6,000 patients, carried out by Johns Hopkins researchers, found that up to two out of every 100 people who come to larger medical centres for treatment were given the wrong diagnosis after histological analysis. The results suggest that second opinion pathology examinations not only prevent errors, but also save lives and money. Consequently, there is an increasing interest among pathology experts in the use of machine vision (or computational diagnosis tools) to reduce the diagnosis error rates by reducing the fallible aspect of human image interpretation. In this thesis, methods for automatically analysing microscopic images of biopsies are investigated.

This introduction chapter first discusses the motivations and objectives of the thesis. The main contributions to knowledge are then detailed. Finally, the outline of the thesis is set out.

1.2 Motivation and Objectives

The human vision system is excellent at performing qualitative tasks, however, it performs less successfully when it comes to quantitative analysis [13]. Relying solely on their eyes can lead pathologists to misdiagnose a sample. MacAulay *et al.* [14] gave the example of a normal-appearing pathologic cell that can be recognised with a quantitative texture analysis of the cell's nucleus, but which stays unnoticed under qualitative observations. Consequently, it is logical to improve the diagnosis accuracy by using computer vision algorithms to quantitatively analyse samples. Computer-aided diagnosis can also assist pathologists in order to reduce the human analysis time, improving efficiency and acting as a second opinion. The addition of computer-based quantitative analysis to the human qualitative interpretation can, furthermore, highly reduce the intra- and inter-observer variability revealed in [11].

The main objective of this thesis is to develop a computerised automatic system for diagnosis of colorectal and prostate tumours using images of biopsy samples. A complete computer-aided diagnosis system consists of two separate phases. First, the unhealthy region needs to be localised, before the system categorises the type of tumour and the grade of the cancer. This thesis focuses on the second phase of the computer-aided diagnosis system. Subsequently, the resulting solution can be integrated into a full computer-aided system using a region segmentation technique or a block-wise image processing method. Numerous investigations for prostate or colorectal tumour classification have already been carried out [15, 16]. However, a large quantity use colour spaces limited to grey-scale or RGB images. In the last decade, an increasing number of studies have used multispectral images [17, 18, 19, 20, 8, 21], which are acquired using a more precise sampling of the light spectrum. This ap-

proach aims at better capturing the spectrum of the reflected light coming from the observed sample, consequently offering more discriminative information. Larsh *et al.* [22] suggested that multispectral imagery has the ability to improve histopathological analysis by capturing patterns that are invisible to the human vision system and to the standard RGB imaging. The research carried out using multispectral imaging has shown promising results and often outperforms the systems using traditional grey-scale or RGB images [15, 16]. However, multispectral images contain a large amount of data which makes them more difficult to process because of increased execution times and problems caused by the curse of dimensionality [19].

The aim of this research is to perform this classification task using multispectral images of colorectal and prostate tumour samples. Two separate datasets divided into four classes each are used in this thesis. The colorectal dataset consists of images acquired on a wide light spectrum, ranging from the visible wavelengths to the infrared wavelengths. This work intends to demonstrate the advantage of using infrared information during image acquisition for the classification of colorectal tumour samples. The thesis plans to investigate, analyse and study different image analysis techniques alongside classification methods, which emphasise on identifying texture features to capture characteristics specific to each type of tumour.

1.3 Contributions

The key original contributions to knowledge of the present thesis are presented as follows:

- A multiclass algorithm for the classification of prostate and colorectal tumours using multispectral imagery is proposed. This sytem is based on a two-dimensional texture extraction combined with a feature selection method. By analysing the performances of this technique on multispectral data and panchromatic images, it was demonstrated that multispectral data show a strong advantage over panchromatic images.

- In order to further utilise the information added by multispectral data, a novel multispectral texture feature based on Local Binary Patterns is introduced. It takes advantage of inter-band spectral information by expending the pixel's neighbourhood considered in Local Binary Pattern features to the spectral dimension. The classification results obtained with this feature are superior to the ones produced by standard texture extractors.

- Powerful classification frameworks are investigated. The bag-of-features scheme uses an image descriptor built as the histogram of texture pattern regions. Regarding the stacked generalisation scheme, texture features are extracted at different scales. Each scale is then fed to a different support vector machine classifier. The output of these classifiers is finally fed to another classifier for the final classification decision. It is demonstrated in this thesis that these frameworks outperform the traditional classifiers for the task at hand.

- Infrared light has not previously been used for histology image classification. This thesis demonstrates the benefits of using infrared information alongside visible light. The performance of the system using multispectral images acquired only in the visible range is compared to results obtained when infrared wavelengths are added. This shows that

the classification accuracy is marginally improved by including infrared information.

- The use of Deep Learning methods is also investigated. More specifically, a convolutional neural network architecture is proposed for biopsy image analysis and classification. The performance of this network is evaluated and compared against the results obtained using features produced by training convolutional neural networks on the ImageNet. Although the transfer learning method has a very high accuracy, the proposed convolutional neural network shows more consistency for the different datasets tested.

1.4 Outline of the Thesis

This thesis is divided into seven chapters.

Chapter 2 provides the essential background in order to understand the data used for this thesis. It presents a biological background on the colon and the prostate gland and describes the anatomy and histology of these organs when healthy. In addition, a detailed explanation of their cancerous and precancerous stages at cellular level is carried out. The acquisition process of colorectal and prostate tumour biopsy images as well as the datasets used in this thesis are depicted.

Chapter 3 focuses on computer vision applied to computer-aided diagnosis problems. It first presents the basics of machine learning systems and describes generic machine learning algorithms. It also addresses the feature extraction phase of computer-aided diagnosis systems. Finally, the state-of-the-art

computer-aided diagnosis systems for colorectal and prostate tumours, including the techniques used for feature extraction and classification of microscopic images are reviewed.

Chapter 4 describes the implementation of four different texture features applied to multispectral microscopic images. Different classifiers are also presented. The chapter demonstrates the usefulness of multispectral data by comparing the performances of different pairings of texture feature and classifier on the datasets manipulated in this work.

Chapter 5 introduces a novel texture feature for multispectral data. It also details two different advanced classification frameworks. The performances of the proposed multispectral texture feature combined with different classification schemes are finally compared and the results analysed. In addition, the usefulness of the infrared spectrum is addressed.

Chapter 6 is concerned with an investigation of deep learning, convolutional neural networks and transfer learning applied to the classification of multispectral biopsy data. The concepts and theory of deep learning are presented with a particular attention paid to convolutional neural networks. In addition, the experiments carried out for the benefit of this thesis are described and the results thoroughly analysed. These experiments compare several network architectures using the datasets described above.

Finally, Chapter 7 presents the conclusions of the thesis and suggests possible further work. It details the main achievements of this thesis, reviewing the results obtained from the different experiments carried out, placing these results in the wider context of the project. Tracks for future investigations are also inspected.

1.5 Conclusion

This introduction chapter presented the context, motivation and objectives of this investigation work. The main contributions to knowledge of the thesis were then addressed. Lastly, the thesis outline was given. The next chapter draws up a biological background on the data used for the project.

Chapter 2

Biological Aspects and Image Acquisition

2.1 Introduction

In this chapter, a biological background on the colon and the prostate gland is discussed. Anatomic and histologic descriptions of the organs in their healthy states will be given before a detailed explanation of the cancerous and precancerous stages at a cellular level. Then, an explanation of the image acquisition process as well as a description of the datasets used in this thesis are discussed.

2.2 Biological Description of the Colon and Prostate Gland

2.2.1 Large Intestine or Colon

Anatomic Description

The large intestine, or colon, is an organ of the human digestive tract. It serves as a reservoir for the fluids evacuated from the small intestine before defecation. This tubular structure of 6 cm diametre and 150 cm length, has for main function to absorb the water and electrolytes of the chyme. Those solutes are transfered to the blood through the membrane of the colon. The colon wall also self lubricates by secreting mucus which facilitates transport of the bowel's content before it can be evacuated by defecation. It also secretes hormones but no digestive enzymes [23].

The rectum is the continuation of the colon and is located just before the anus. It has a similar tissue structure to the colon with folds comparably to the *plicae circularis* present in the small intestine. Figure 2.1 shows the anatomy of the colon and rectum.

Histologic Description

Figure 2.2, shows a slide of the colon tissues with caption. The mucosa or mucous membrane is a tissue type present in different internal organs of the body. It is composed of the epithelium and the *lamina pripria*. The epithelium is a single layer of column-shaped epithelial cells that has a thin brush border. The *lamina propria* is a connective tissue that has a rich vascular and lymphatic

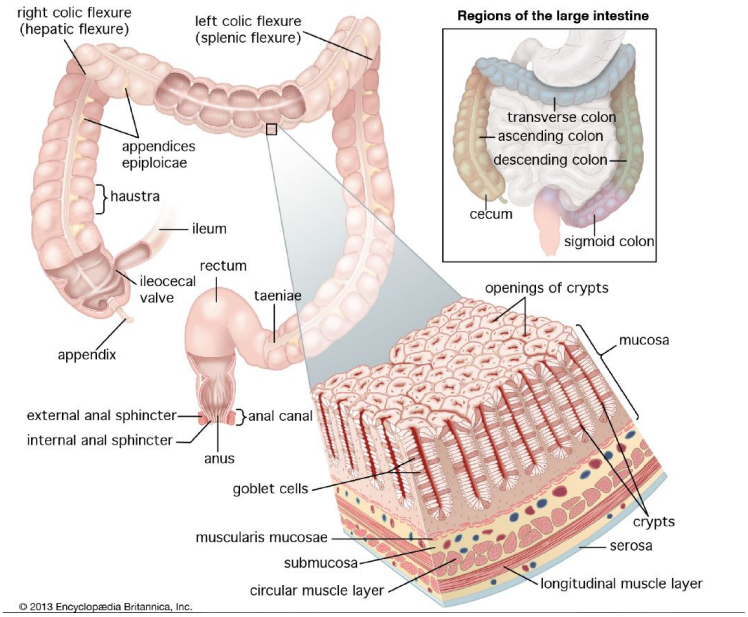


Figure 2.1: Structures of the human large intestine, rectum, and anus. [24]

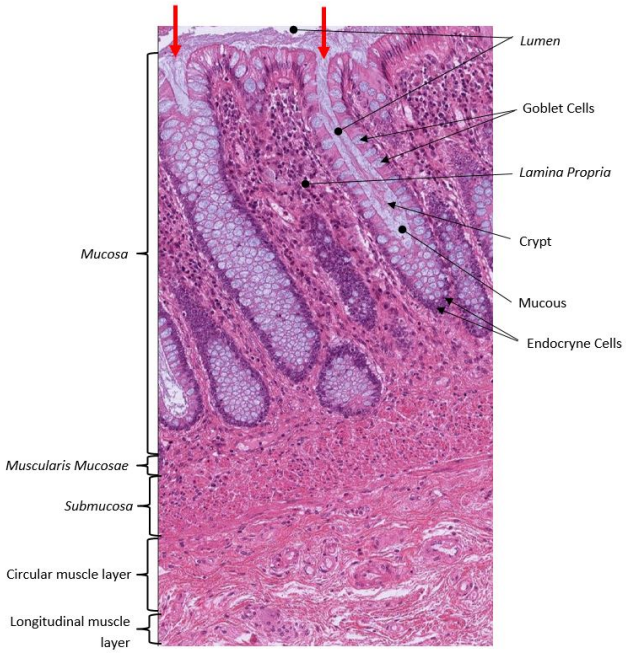


Figure 2.2: Histology of a slide of the colon with a 600× magnification. The red arrows show the crypts openings.

network, which absorbs the digestive products [25, 26]. The mucosa displays deep crypts, the crypts of Lieberkühn, which are straight and unbranched. The walls of these crypts are lined with a great number of goblet cells which are recognisable by their larger size and paler colour. The crypts bases are covered with undifferentiated cells and endocrine cells which are typically smaller and darker. Mucus is secreted by the goblet cells and is usually present on the walls of the crypts [23]. The *lumen*, which is the inside of the bowel, extends to the inside of the crypts of the mucosa. A *muscularis mucosae* layer is present immediately at the base of the crypts. It consists of separate inner circular and outer longitudinal layers of muscles.

2.2.2 Prostate Gland

Anatomic Description

The prostate gland is a chestnut-shaped organ of the male reproductive system. With a 4 cm diametre at the broadest area, it is located in the pelvis, inferior to the urinary bladder (see Figure 2.3). Its main function is to secrete a fluid that is added to the seminal fluid during ejaculation [25].The prostate consists of 30 to 50 tubular or sack-like glands organised into three concentric layers: an inner mucosal layer, an intermediate submucosal layer, and a peripheral layer where the main prostatic glands are located.

Histologic Description

The secretory portion of the glands consists of a simple layer of pseudostratified columnar epithelium supported by the non secretory *stroma*. At the centre

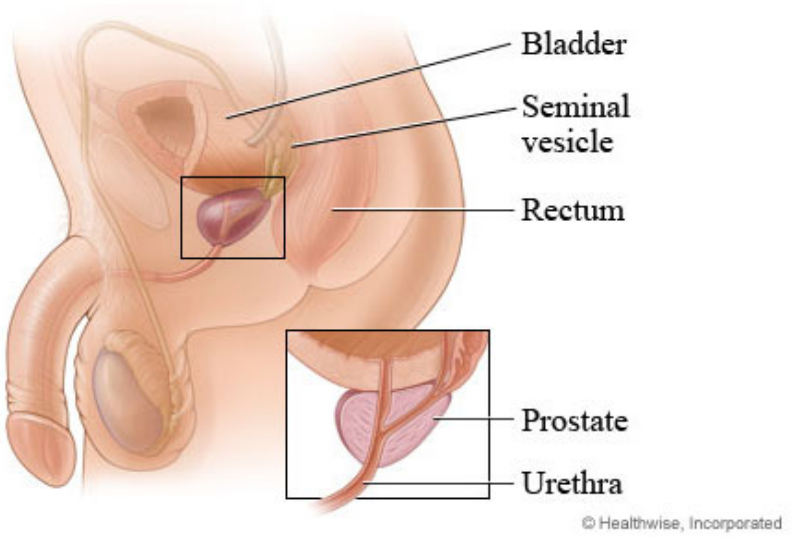


Figure 2.3: Anatomy of the prostate gland. ©1995-2015 Healthwise, Incorporated.

of the glands, the *lumen* of the alveoli is enclosed by the epithelium. The *stroma* is made up of the *lamina propria* which is intimetely intermingled with a layer of smooth muscle called the *muscularis mucosae* as can be seen on Figure 2.4. The *muscularis mucosae* can be recognised from the *lamina propria* by a more intense staining. The upper inset of Figure 2.4 shows that there is no clear outlined layers of smooth muscle in the prostate; instead, it is randomly organised throughout the *stroma*.

2.2.3 Polyps, Tumours, and Cancer

Anatomic Characteristics

As described in Section 2.2, both the prostate gland and the colorectum have a similar tissue organisation with the tubular glandular mucosa – composed of epithelium and lamina propria – being their main functional tissue. This quality means that they are subject to developing the same types of tumours

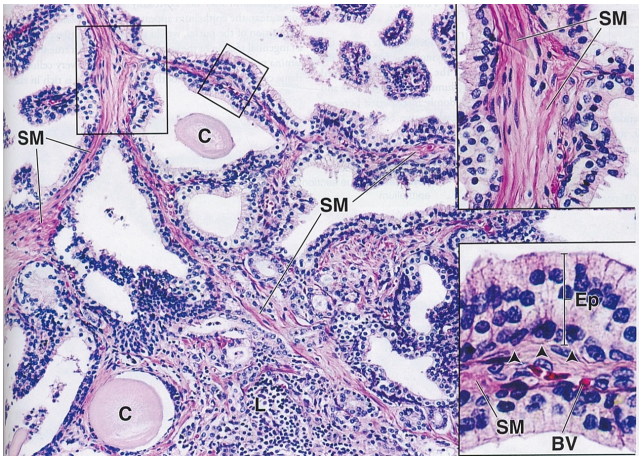


Figure 2.4: Histology of a slide of the prostate gland ×178; upper inset corresponding to the larger rectangle (×350); lower inset corresponding to the smaller rectangle (×650). The lamina propria and smooth muscle (SM) are visible adjacent to the secretory epithelium (Ep) and in the non secretory areas. Prostatic concretions (C) – aggregations of dead epithelial cells and precipitated secretions – are observable in the *lumina* of the alveoli. In the lower inset, the basal cells (arrowheards) are seen along taller columnar secretory cells showing the pseudostratified nature of the epithelium in the prostate gland. A blood vessel (BV) is recognisable by its red colour. Lymphocytes are visible at the lower border of the main image indicating an inflammation of the prostate gland. [25]

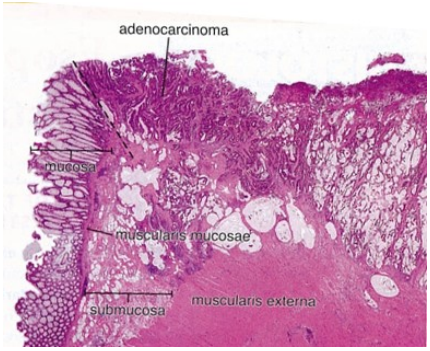


Figure 2.5: Low magnification photography of a microscopic view of an andenocarcinoma of the colon. (×120) [25]

and cancers. Carcinomas are the most common type of malignant tumours, they derive from epithelial cells [27]. Carcinomas are called adenocarcinomas when derived from glandular tissues – which is the case for both of the organs studied in this work.

All growths are not necessarily malignant and benign polyps can occur [28]. They usually are non-cancerous growths of the mucosa into the lumen and can be of different types. Although most polyps are completely benign like the hyperplastic polyps or hyperplasia, some types of polyp can transform into an adenocarcinoma and can as such be considered as a pre-cancerous stage. They are called adenoma and can be tubular or villous depending on their growth patterns [29]. Hyperplastic polyps are characterised by an increase in the number of cells resulting in an increased size of the tissue due to an enhanced cell division. In contrast to an adenoma or a carcinoma, the division rate in a hyperplastic polyp returns to normal as soon as the stimulus is removed.

Histologic Characteristics

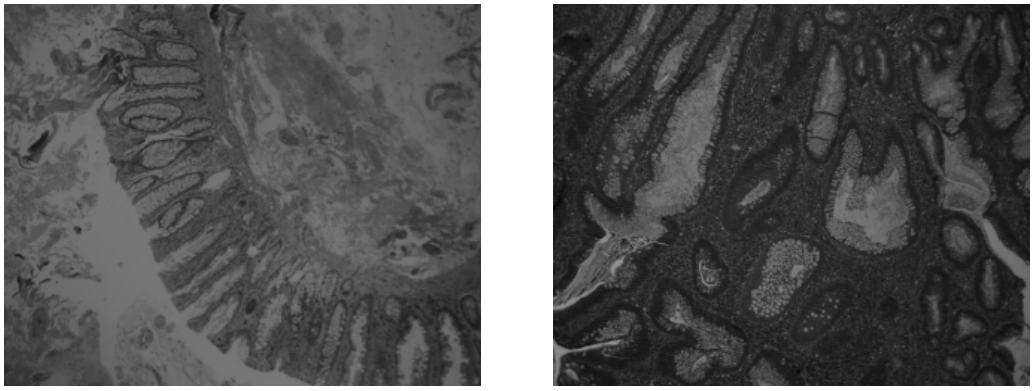
An adenocarcinoma has easily recognisable features. Unlike the well organised healthy tissues described in Section 2.2 and visible at the left of Figure 2.5,

the lesioned tissue shows an irregular pattern of glands and is hyperchromatic, meaning it is intensively stained with the hematoxylin staining agent – top left of Figure 2.5. Moreover, the muscle fibres from the muscularis mucosa are observable among the glands.

The grade of an adenomatous polyp is assessed by the degree of dysplasia. This is to describe how differentiated the cells are. A low-grade dysplasia means that, similarly to a healthy tissue, the cells are well differentiated, each of them having their specific shape and function. On the other end of the spectrum, a high-grade dysplasia means that the polyp is more cancerous-like with irregular patterns of glands, hyperchromatism, and without production of mucus [25].

Figure 2.6 shows a comparison between a low grade adenomatous polyp and a hyperplastic polyp. Both samples of Figure 2.6a and 2.6b have hyperchromatism and the crypts are more crowded with goblet cells than in a normal tissue (see Figure 2.2). However, the main difference between the two lies in the shape of the crypts. On Figure 2.6a, the crypts have a close to normal straight simple tubular crypts. On the other hand, Figure 2.6b, shows an early case of deformation of the crytps with the appearance of branches and folds inside them.

In the specific case of prostate cancer, the Gleason grading system has been the main tool to recognise the stages of the adenocarcinoma tumours [30, 31, 32]. It ranks patterns, observable at low magnification, from the most differentiated (Gleason pattern 1) to the least differentiated cells (Gleason pattern 5), as shown on Figure 2.7. Most of the time, several patterns can be observed on prostate carcinomas. For this reason, a primary and secondary Gleason patterns are defined as the two most prevalent patterns on the tissue. The



(a) Photography of a microscopic view of a hyperplastic polyp of the colon. ×100

(b) Photography of a microscopic view of a low grade tubular adenoma of the colon. ×200

Figure 2.6: Comparison of microscopic views of hyperplasia and adenoma for the colon.

final grading Gleason score is given by the sum of these two predominant patterns' number. It ranges from 2 (1+1) to 10 (5+5) [33] and will determine the adequate treatment for a patient. However, Gleason patterns 1 and 2 are not currently included in the calculation of the final Gleason score grading anymore, making 6 (3+3) the lowest possible score. This is why most datasets used for automatic diagnosis are divided into classes corresponding to Gleason patterns ranging from 3 to 5. Another diagnosis system has been defined in [30]. It groups Gleason grades into new categories as shown on Table 2.1. This is due to a difference in the Gleason grade 3+4 and 4+3 which should lead to different treatments. This means that differentiating the Gleason patterns 3 and 4 are particularly important tasks, whereas the differences between patterns 4 and 5 have less impact on the final diagnosis.

This histologic description of the different types of tumour and polyps highlights how difficult it can be to discriminate between them. Every case sits somewhere on a spectrum. Limits between different categories can sometimes be hard to determine, making specialists disagree over some samples. Thus,

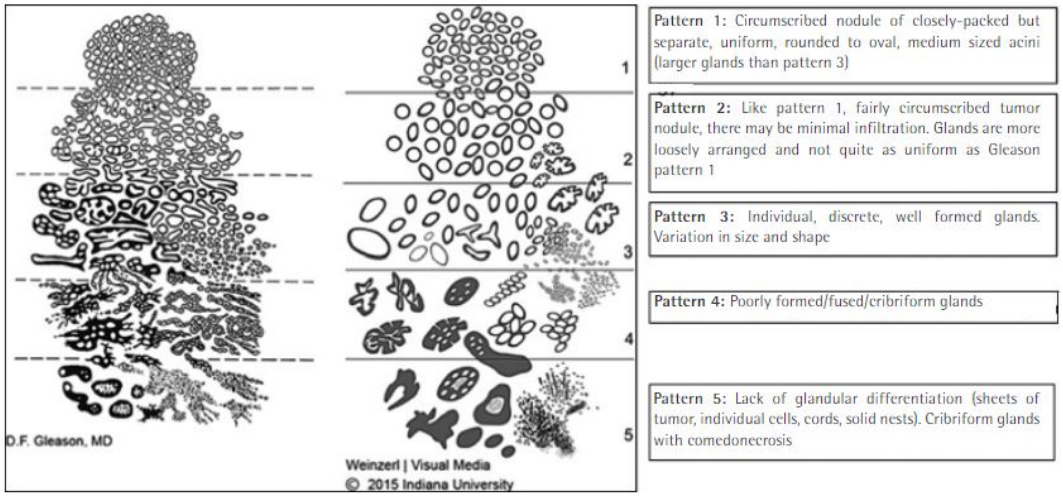


Figure 2.7: Gleason histologic patterns of the prostatic adenocarcinoma schematic diagrams. Original version (left) and 2015 Modified International Society of Urological Pathology (right). [33]

Table 2.1: Gleason grade groups

Gleason grade group	Gleason score
1	≤ 6
2	$3+4=7$
3	$4+3=7$
4	$4+4=8$, $3+5=8$, $5+3=8$
5	9-10

the use of a machine learning algorithm could help give them a second opinion for the litigious cases.

2.3 Optical Microscopy

Conventional transmitted light widefield microscopy was used for acquiring the images of this study's datasets. It consists of a collecting lens (also called the objective lens), an ocular lens (or eyepiece lens), a detector (either the human eye or a Charge-Coupled Device (CCD) camera), and a light source. It is suitable for inspecting thin and transparent samples. The sample is usually stained to increase the contrast of their structure of interest compared to the rest of the sample. The light coming from the source is reflected on the sample and travels trough the lenses before hitting the detector. the inverted magnification of the object is formed at the imaging plane and observable by the detector.

2.4 Sample Preparation

Sample preparation is usually carried out by a pathologist following a precise protocol.

2.4.1 Sample Collection

For the collection of the prostate dataset, the samples were provided by the Institute of Pathological Anatomy and Histopathology, University of Ancona, Italy. They were taken from prostate ablations in order to have a full section of the tissue.

The colorectal dataset was acquired from samples provided by Dr. Rafif from Al-Ahli Hospital, Doha, Qatar. They were extracted from colorectal biopsies.

2.4.2 Section Preparation

The following section preparation stages are the same for both datasets.

After they were collected from patients, the ablated tissues were fixed in a "life-like" state by being immersed into formalin for 6 to 12 hours. Formalin is a fixing agent that stops enzyme activity, kills microorganisms while keeping the molecular structure intact in order to allow the appropriate reaction with the staining agents. Tissues then go through a series of ethanol and xylene baths to be dehydrated before being embedded in molds filled with wax (which is immiscible with water, hence the dehydration phase). After the wax has solidified, the "block" of wax embedded with the tissue sample is sectioned into 3 to 5 μm -thick slices to make sure only one layer of cells is present on each section. These sections are then floated out on a warm water bath so they can flatten, picked up onto a glass microscope slide and dried [17].

Most of the cells on the tissue are colourless at this stage. In order to reveal the tissue structures, the samples need to undergo a staining stage. The most widely used staining agents are Haematoxylin & Eosin (H&E) stains. When using H&E stains on a tissue sample, its nuclei take a dark blue colour while its cytoplasms and other components show different shades of pink. The slides are then covered with a glass cover and are ready for microscopic observation.

2.5 Multispectral Imaging

Traditional cameras function following a RGB model mimicking the human eye's vision mechanism. Visible light is captured by three types of sensors (which are cone cells in the retina). The sensor's (respectively cone cell) re-

sponse to a particular wavelength λ is given by its responsivity function $R(\lambda)$. Each type of sensor has a response in the visible spectrum of light and its responsivity is positive on a limited range and equals to zero elsewhere. The bandwidth in which the sensor's responsivity is positive is different for each type of sensor: the first type will respond to wavelengths corresponding to blue (400-500 nm), second type to green (450-630 nm), and the third one to red (500-700 nm). In a vision system, the emitted light is reflected by the observed object and reaches the sensor or eye. The spectrum of the reflected light, called reflectance, varies depending on the colour of the object. The system's output is a projection of this reflectance on the different colour sensors. As a result, two objects with different reflectances could give similar outputs and be perceived as having the same colour.

For this reason, a more accurate approximation of the light spectrum is done using many more sensors than the usual three used in the RGB systems, with each sensor having a narrow bandwidth. Depending on the bandwidth resolution, the number of spectral bands can be varied to capture more precise information on the light spectrum: the smaller the bandwidth resolution, the larger the number of spectral bands and the more accurate the approximation is. As suggested by Larsh *et al.* [22], a more accurate approximation of the reflectance allows for an improved histological analysis by capturing patterns that are invisible to the human vision system and the standard RGB imagery.

The resulting image is a cube of data consisting of several gray level spectral images, each of them being acquired using a spectral filter centred on a particular wavelength of the electromagnetic spectrum. Consequently, this three-dimensional image has two spatial dimensions and one spectral dimension.

An object's reflectance goes beyond the visible spectrum and some sensors are designed to capture wavelengths that are in the infrared (IR) spectrum. A multispectral imaging system can therefore be adapted to an extended spectrum, capturing images throughout the visible and the IR ranges of the electromagnetic spectrum.

2.5.1 Imaging System and Equipment

Both datasets used in this thesis were acquired using the optical system described in Section 2.3, except that a Liquid Crystal Tunable Filter (LCTF) was placed between the microscope and the camera to simulate the behavior of spectral sensors.

The prostate cancer was acquired using a ™VARISPEC LCTF with a spectral range of 400-720 nm.

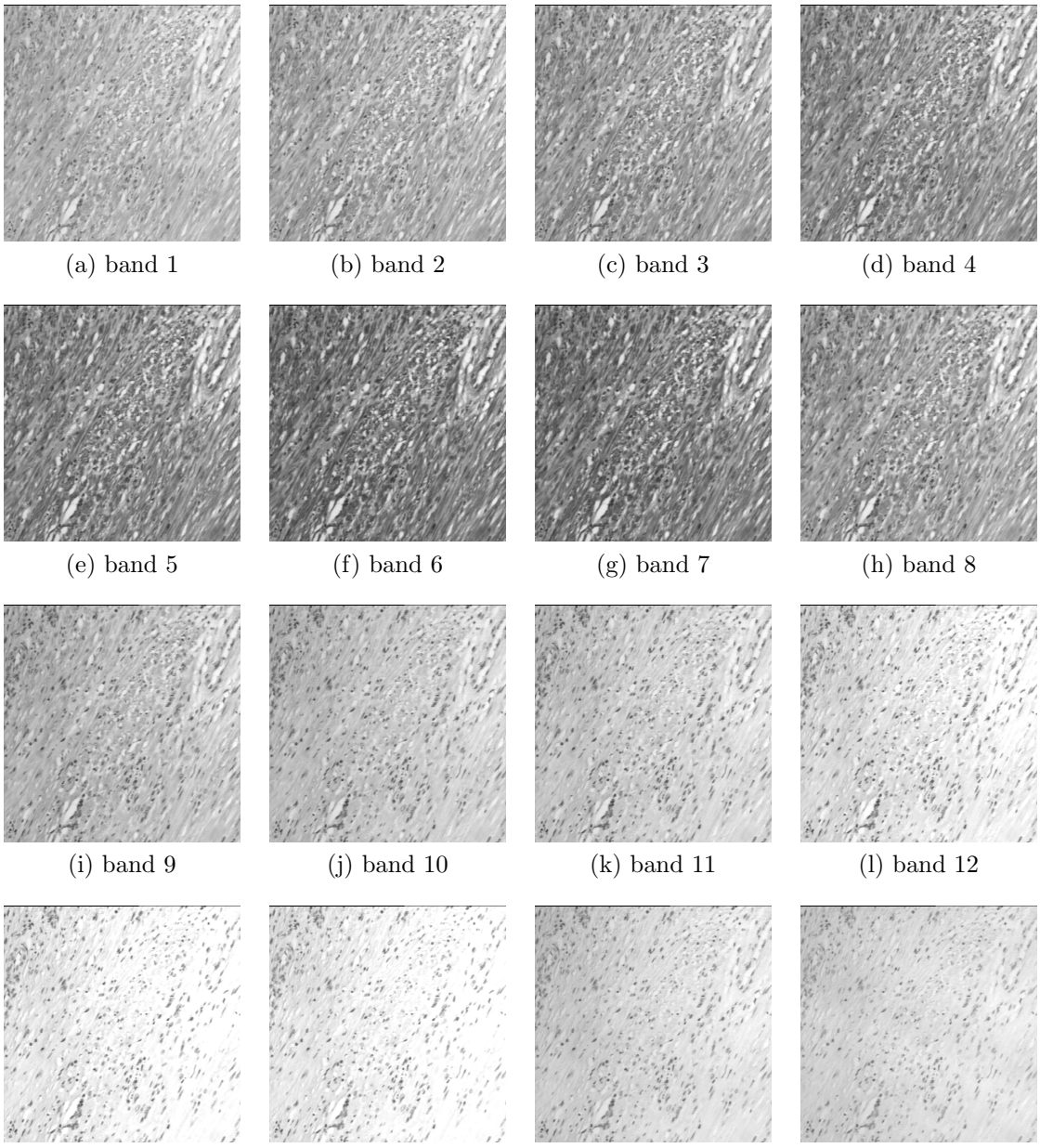
For the colorectal dataset, images were acquired in visible light and IR. The microscope was equipped with a halogen illumination emitting in the visible and the IR spectra. The camera used was a XENICS CHEETAH with a spectral range of 400-1700nm and a resolution of 320×265 pixels. Two separate ™VARISPEC LCTF were needed to cover the whole spectrum required: one for the visible spectrum (400-720 nm) and another one for the IR (850-1800 nm).

2.6 Datasets Decription

Two datasets were used for the purpose of this thesis.

The first one, referred to as **prostate dataset**, was used in previous works by Tahir *et al.* [19]. It consists of 512 different multispectral prostate tumour tissue images of size 128 × 128. The images are taken at 16 spectral channels (from 500 to 650 nm) and at x40 magnification power. The samples are evaluated by two highly experienced independent pathologists and labeled into four classes: 128 cases of Stroma (Str), which is normal muscular tissue, 128 cases of Benign Prostatic Hyperplasia (BPH), a benign condition, 128 cases of Prostatic Intraepithelial Neoplasia (PIN), a pre-cancerous stage, and 128 cases of Prostatic Carcinoma (PCa), an abnormal tissue development corresponding to cancer. Samples of the prostate dataset are shown in Figures 2.8, 2.9, 2.10, 2.11.

The second dataset, referred to as **colorectal dataset**, is composed with multispectral colorectal histology data with a (x40) magnification power. This dataset was developed by the University of Qatar with the collaboration of the Al-Ahli Hospital, Doha. It is split into 4 classes, each of them composed of 40 images. The images are acquired on a wider spectrum than in the first dataset as it is spread on the visible (Vis) and infrared (IR) ranges of the electromagnetic spectrum with an interval of 23 nm between each wavelength. That is to say, in the visible range, the wavelength interval is 23 nm starting from 465 nm to 695 nm and in the IR range, the wavelength interval is also 23 nm and ranges from 900 nm to 1590 nm. The spacial size is 128 × 160. The 4 classes are: Carcinoma (Ca), containing the images of cancerous colon biopsies, Tubular Adenoma (TA), a pre-cancerous stage, Hyperplastic Polyp (HP), a benign polyp and No Remarkable Pathology (NRP). Samples of the colorectal dataset are shown in Figures 2.12, 2.13, 2.14, 2.15.



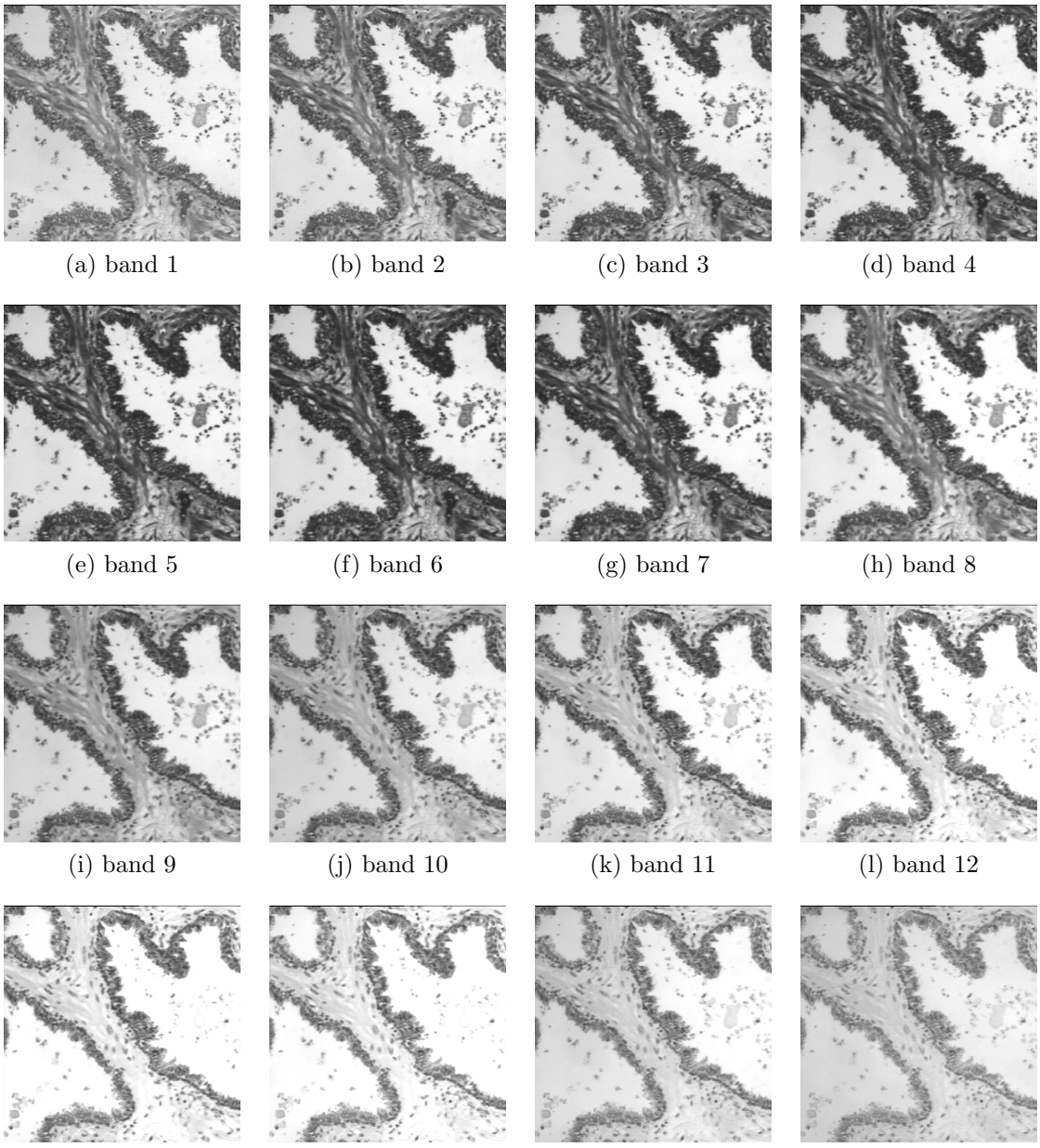
(a) band 1 (b) band 2 (c) band 3 (d) band 4

(e) band 5 (f) band 6 (g) band 7 (h) band 8

(i) band 9 (j) band 10 (k) band 11 (l) band 12

(m) band 13 (n) band 14 (o) band 15 (p) band 16

Figure 2.8: An extract from the spectral bands of a sample of class Str taken from the prostate dataset



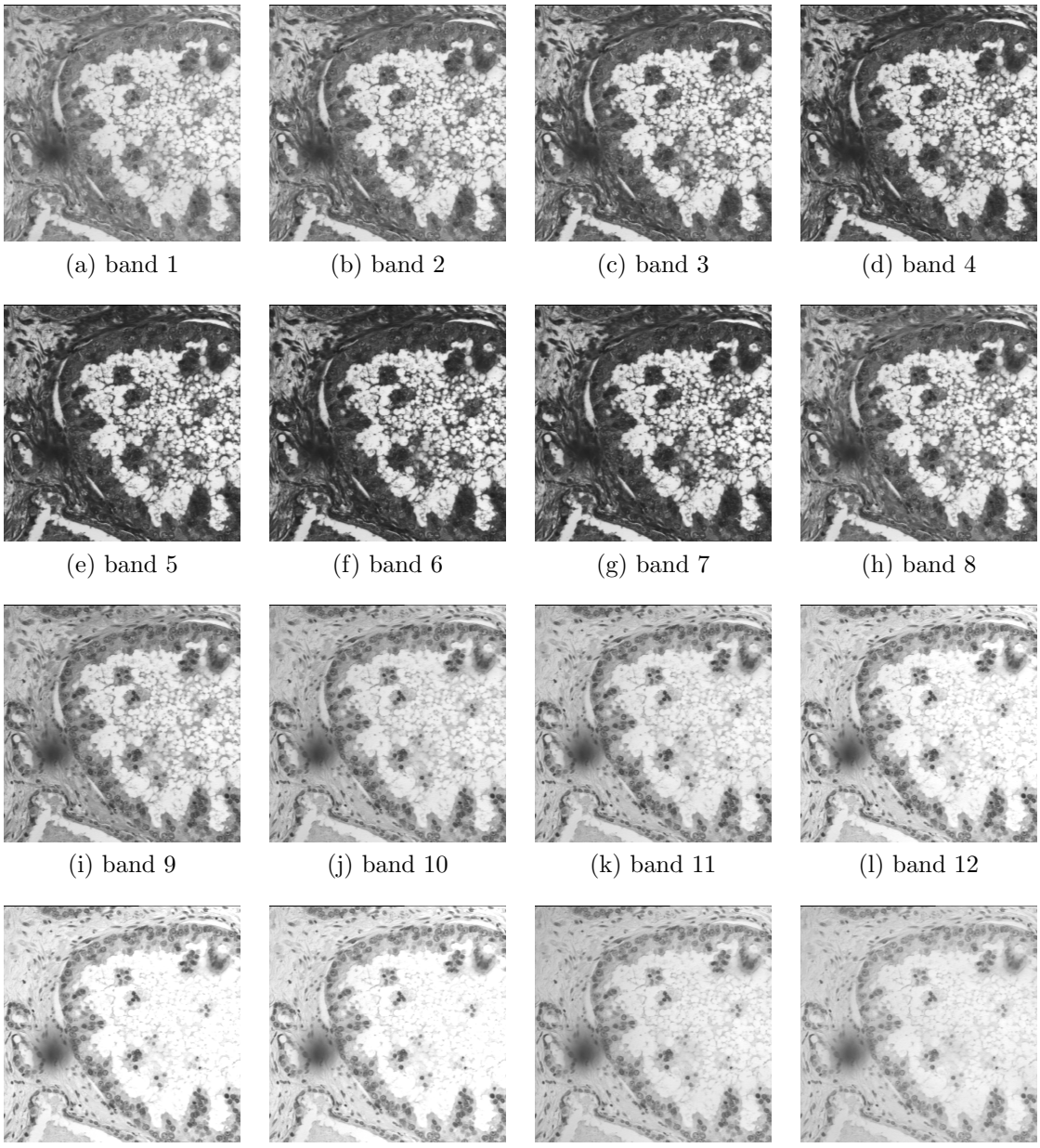
(a) band 1 (b) band 2 (c) band 3 (d) band 4

(e) band 5 (f) band 6 (g) band 7 (h) band 8

(i) band 9 (j) band 10 (k) band 11 (l) band 12

(m) band 13 (n) band 14 (o) band 15 (p) band 16

Figure 2.9: An extract from the spectral bands of a sample of class BPH taken from the prostate dataset



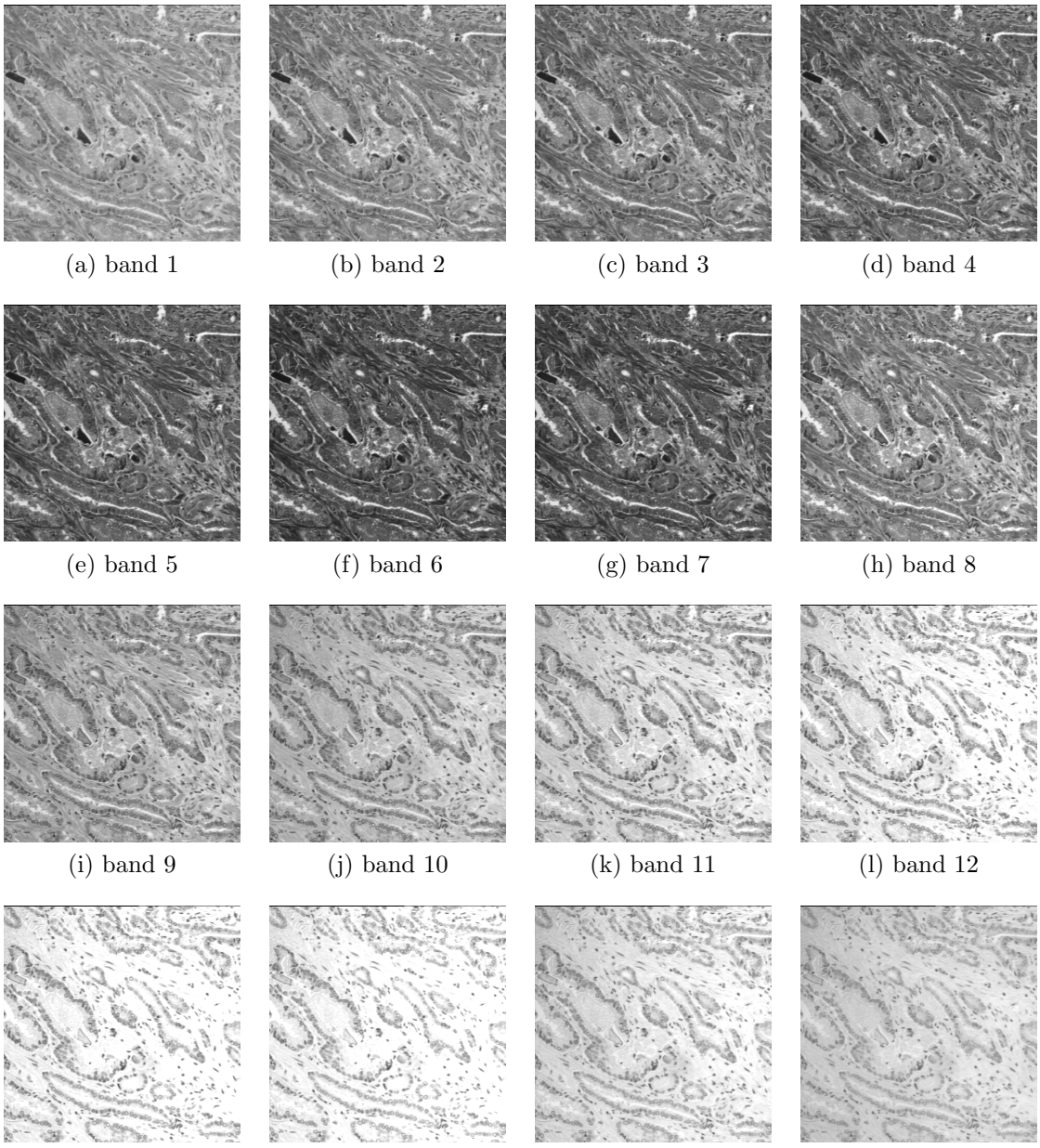
(a) band 1 (b) band 2 (c) band 3 (d) band 4

(e) band 5 (f) band 6 (g) band 7 (h) band 8

(i) band 9 (j) band 10 (k) band 11 (l) band 12

(m) band 13 (n) band 14 (o) band 15 (p) band 16

Figure 2.10: An extract from the spectral bands of a sample of class PIN taken from the prostate dataset



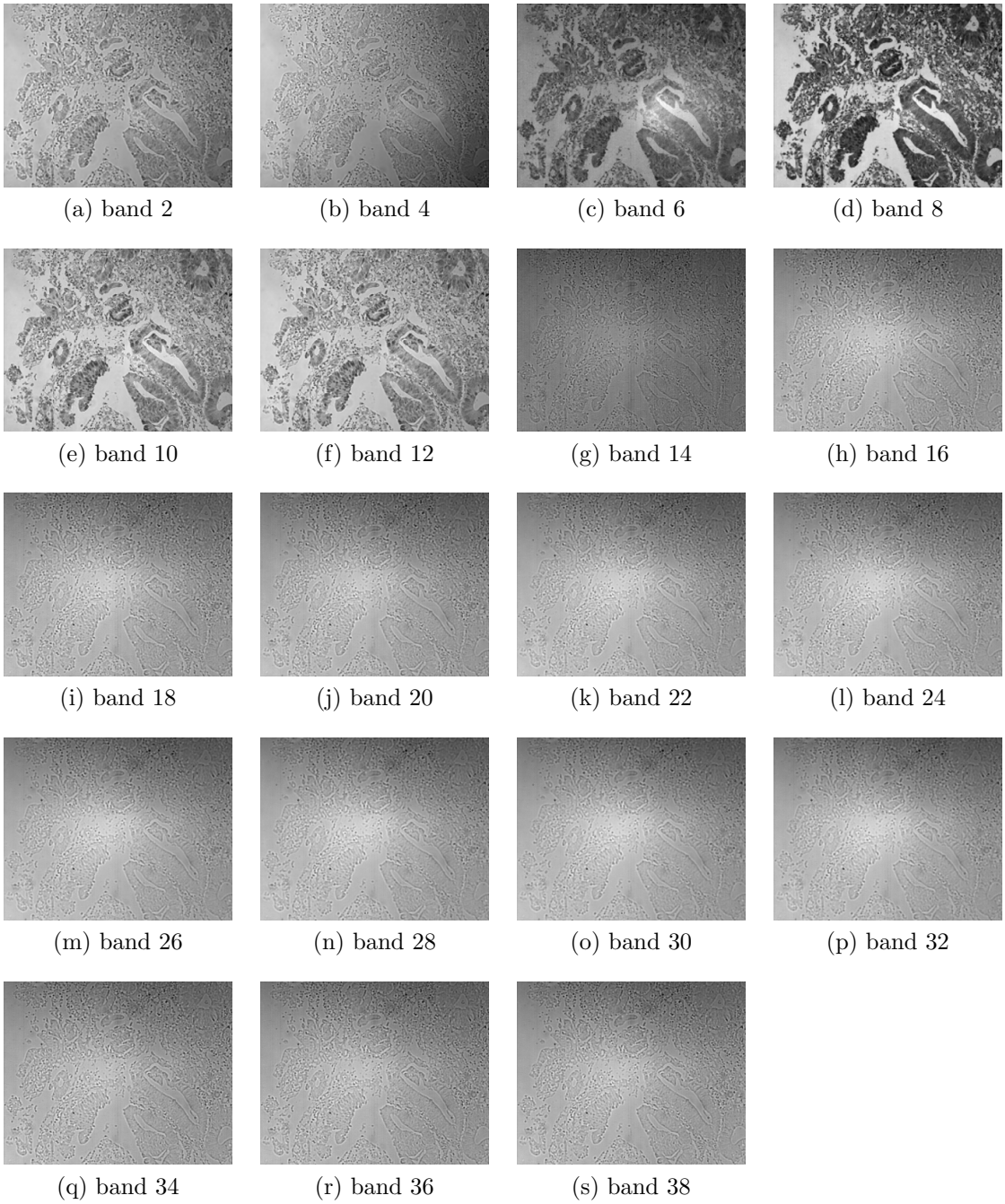
(a) band 1 (b) band 2 (c) band 3 (d) band 4

(e) band 5 (f) band 6 (g) band 7 (h) band 8

(i) band 9 (j) band 10 (k) band 11 (l) band 12

(m) band 13 (n) band 14 (o) band 15 (p) band 16

Figure 2.11: An extract from the spectral bands of a sample of class PCa taken from the prostate dataset



(a) band 2 (b) band 4 (c) band 6 (d) band 8

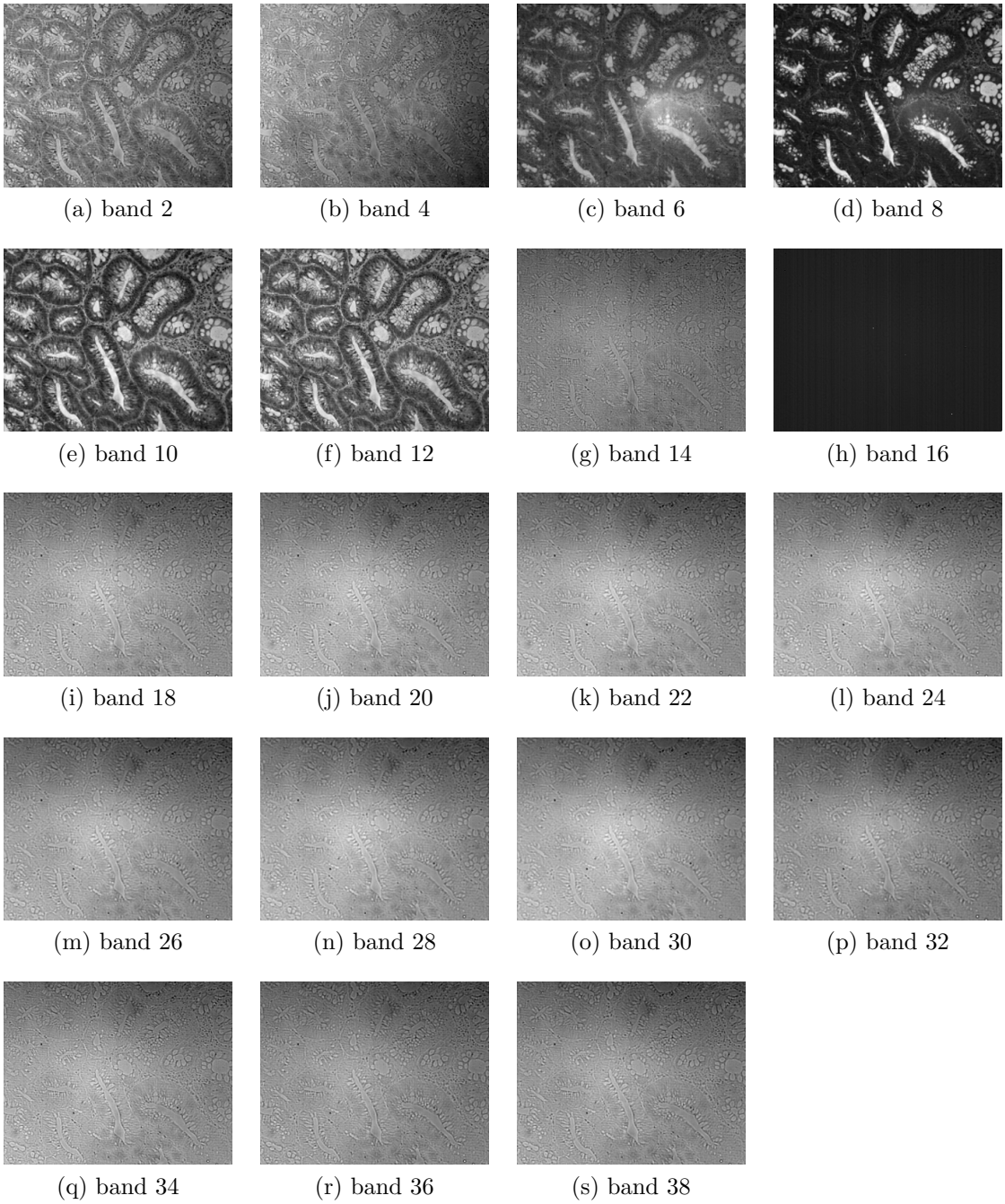
(e) band 10 (f) band 12 (g) band 14 (h) band 16

(i) band 18 (j) band 20 (k) band 22 (l) band 24

(m) band 26 (n) band 28 (o) band 30 (p) band 32

(q) band 34 (r) band 36 (s) band 38

Figure 2.12: Spectral bands of a sample of class Ca taken from the colorectal dataset



(a) band 2 (b) band 4 (c) band 6 (d) band 8

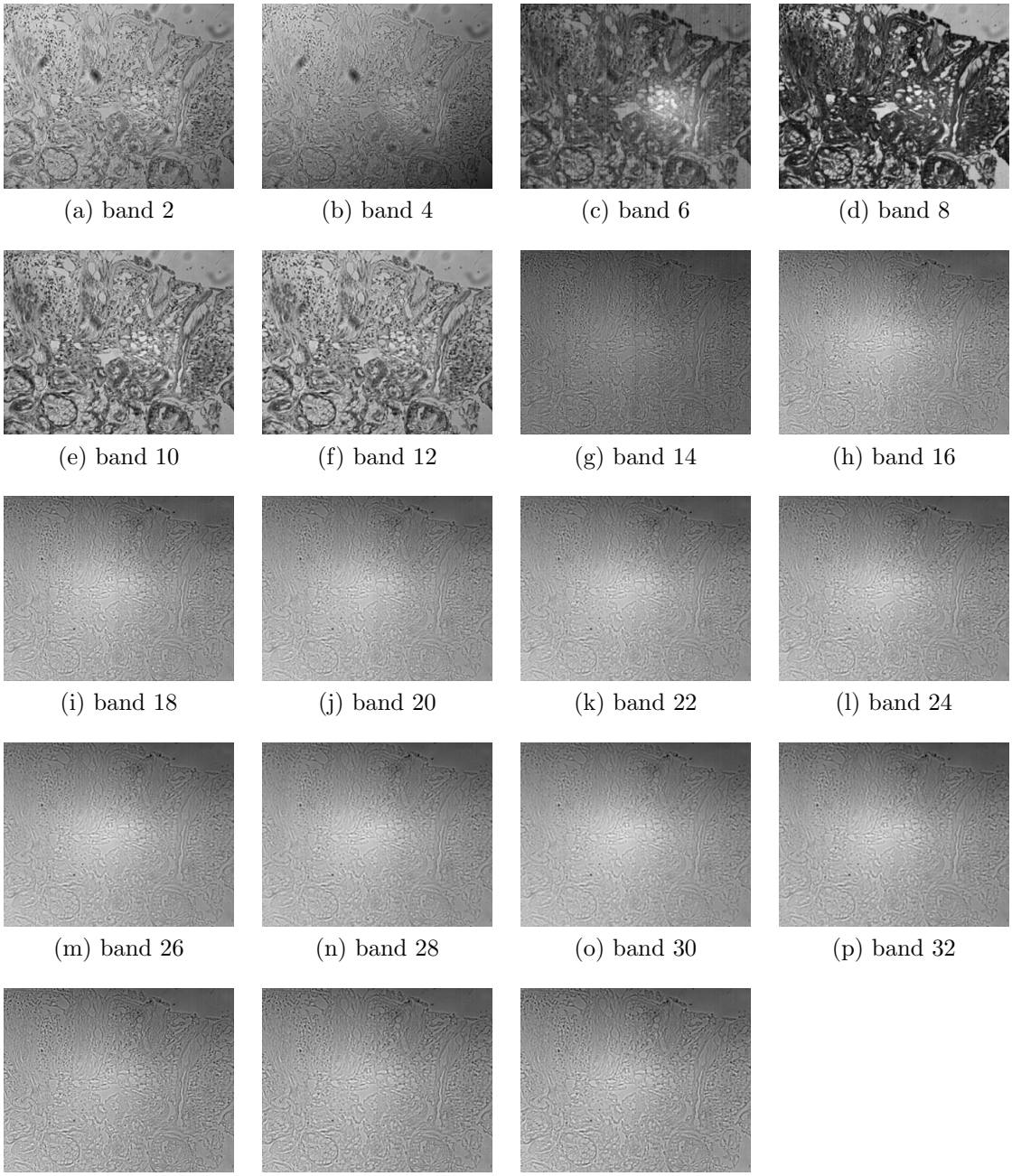
(e) band 10 (f) band 12 (g) band 14 (h) band 16

(i) band 18 (j) band 20 (k) band 22 (l) band 24

(m) band 26 (n) band 28 (o) band 30 (p) band 32

(q) band 34 (r) band 36 (s) band 38

Figure 2.13: Spectral bands of a sample of class Ta taken from the colorectal dataset



(a) band 2 (b) band 4 (c) band 6 (d) band 8

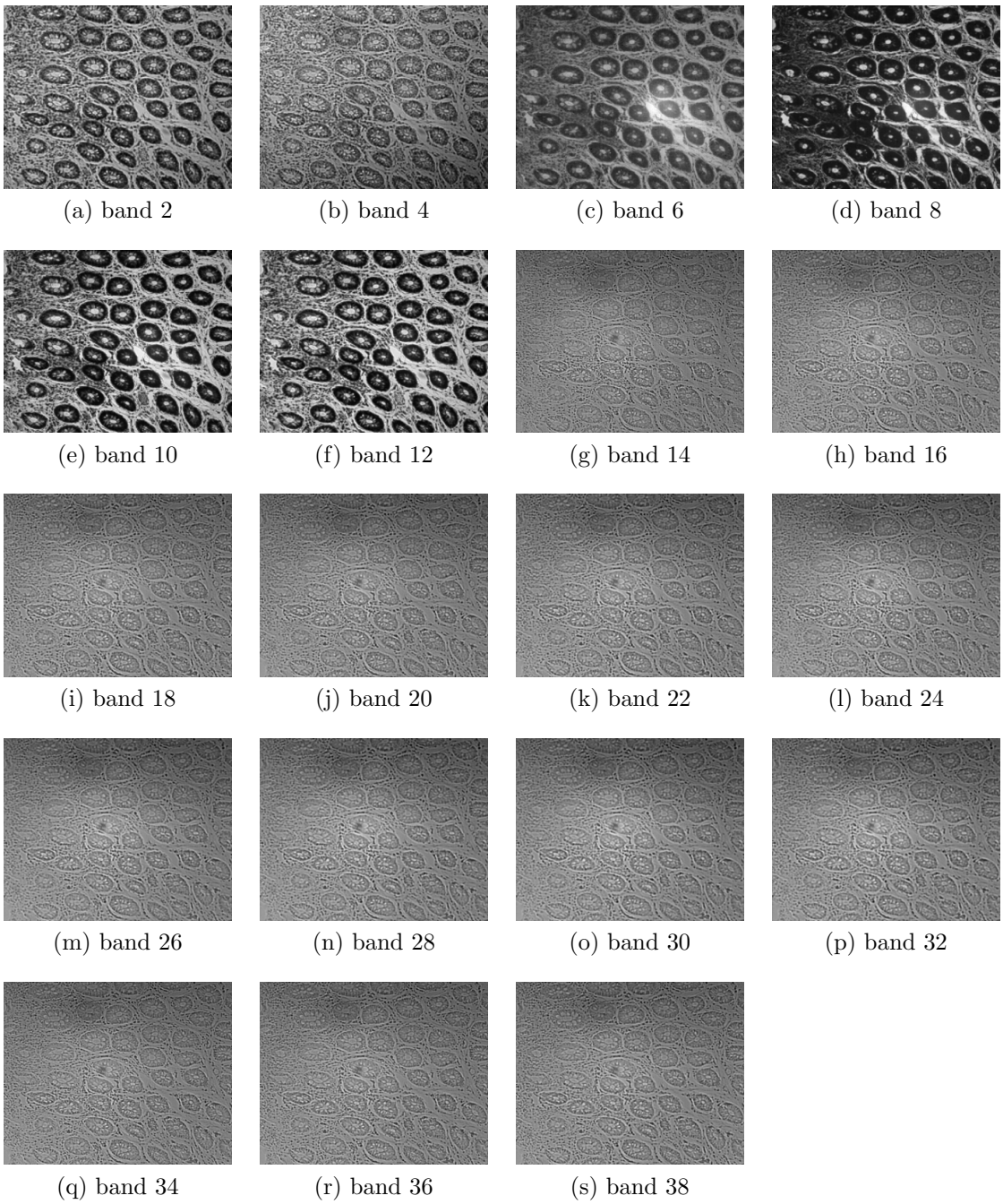
(e) band 10 (f) band 12 (g) band 14 (h) band 16

(i) band 18 (j) band 20 (k) band 22 (l) band 24

(m) band 26 (n) band 28 (o) band 30 (p) band 32

(q) band 34 (r) band 36 (s) band 38

Figure 2.14: Spectral bands of a sample of class HP taken from the colorectal dataset



(a) band 2 (b) band 4 (c) band 6 (d) band 8

(e) band 10 (f) band 12 (g) band 14 (h) band 16

(i) band 18 (j) band 20 (k) band 22 (l) band 24

(m) band 26 (n) band 28 (o) band 30 (p) band 32

(q) band 34 (r) band 36 (s) band 38

Figure 2.15: Spectral bands of a sample of class NRP taken from the colorectal dataset

2.7 Conclusion

In this chapter, we gave a broad description of the biological aspects of a the prostate and the colon. The anatomic and histological characteristics of the different types of tumour and cancer found in these organs were also explained. Then, the image of the tissue sample acquisition system was detailed. Especially, we described the samples extraction, processing, and staining. We also explained the acquisition system and its components. Finally, the two datasets used in this study were described.

In the next chapter, an insight on machine learning and computer-aided diagnosis systems (CADS) will be given. We will also give a detailed state of the art of automatic diagnosis systems for prostate and colorectal cancers.

Chapter 3

Machine Learning and Computer-Aided Colorectal and Prostate Cancer Diagnosis Systems

3.1 Introduction

This chapter discusses the basics of machine learning systems in general and describes generic learning algorithms. It also describes texture feature extraction. It then thoroughly reviews the state-of-the-art in the field of CADS for colorectal and prostate cancers.

3.2 Machine Learning Basics

Goodfellow *et al.* [34] define machine learning as "a form of applied statistics with increased emphasis on the use of computers to statistically estimate complicated functions and a decreased emphasis on proving confidence intervals around these functions" and "an algorithm that is able to learn from data." Mitchell *et al.* [35] provide a definition for learning in this context, "a computer program is said to learn from experience E with respects to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."

3.2.1 Definition of Learning Algorithms in the Context of Classification

The task T Thanks to machine learning, we are able to address tasks that cannot be solved with a rigid program conceived by humans. In the previous definition, the learning process is not the task T but rather a method to be able to complete this task. In the case of this thesis, the task at hand is a classification task. However in other situations, this task could also be a regression, a translation or a transcription task [34]. In a classification task, the aim is to determine the category of an input. The input, $\mathbf{x} \in \mathbb{R}^n$, is called an example and it is a collection of features that have been computed from the object to classify. To solve the task, the learning algorithm has to construct a function $f : \mathbb{R}^n \mapsto \{1, \ldots, k\}$, where k is the number of categories – or classes. The model thus assigns to an example $\mathbf{x}$ a class $y \in \{1, \ldots, k\}$, $y = f(\mathbf{x})$.

The performance measure P In order to assess the performance of a machine learning system, a quantitative measure P is used. For a classification task, this performance measure often is the accuracy:

$$Accuracy = \frac{\text{number correctly classified examples}}{\text{total number of examples}}.$$

However, classification tasks are often asymmetrical, especially in a medical context. For instance, detecting a cancer when there is not any does not have the same impact as not detecting a cancer when there is one. In this case, the accuracy does not give enough insight on the classifier's performance. It does not tell how the misclassified examples are split between classes. In a binary classification problem, True Positives, TP (resp. True Negatives, TN), are the examples that were correctly classified in the positive (resp. negative) class. Examples classified as negative (resp. postive) when they were in fact positive (resp. negative) are called False Negatives, FN (resp. False Positives, FP) [36]. Two new metrics are therefore defined:

$$Precision = \frac{TP}{TP + FP}, \tag{3.1}$$

The *Precision* is the fraction of correctly classified positives in all the examples classified as positive by the model.

$$Recall = \frac{TP}{TP + FN}, \tag{3.2}$$

The *Recall* is the fraction of correctly classified positives in all the examples that should have been classified as positive.

Another performance measure very often used is the receiver operating characteristic curve (or ROC curve) [36]. This curve displays the True Positive Rate (TPR) on the y-axis and the False Positive rate (FPR) on the x-axis. It shows the variation of the two rates depending on the value of the discrimination threshold. This threshold is the model's output value from which an example is classified as positive. In order to draw the ROC curve, the discrimination threshold is varied from 0 to 1 and for each threshold value, the TPR and FPR are computed, resulting in a point on the ROC curve. Points are then linked in order to create the curve. The point at the right end (resp. left end) of the curve corresponds to classifying every example as a positive (resp. negative). A perfect classifier would be as much as possible in the upper left corner of the graph. Consequently, the left part of the ROC and its steepness near the origin is an important factor to consider when looking at a ROC curve. A widely used metric to compare two ROC curves is the area under curve (AUC) which gives an average value of the classifier performance but does not substitute the curve itself. With a perfect classifier, the AUC equals 1, while for a random binary classifier, the AUC equals 0.5. Figure 3.1 shows an example of ROC curve and its AUC.

The performance measures how the algorithm performs with new, previously unseen data, to have a better idea of how it will perform on real world data. The collection of examples, called dataset, is therefore split into a training set, used to build the aforementioned f function, and a test set, used to evaluate the model's performance measures. This is called the data-generating process. The generalisation is the performance of the system on the test set.

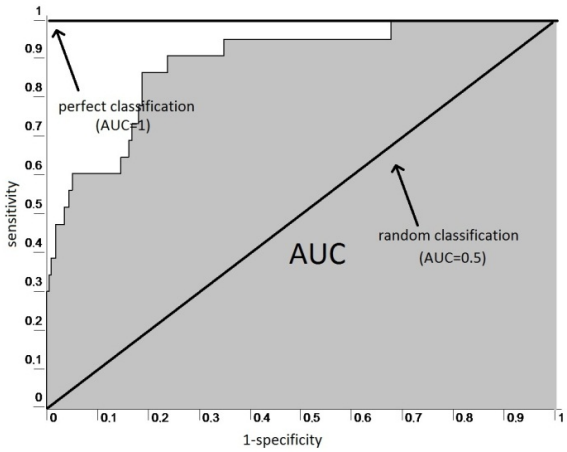


Figure 3.1: Example of a ROC curve and its AUC. The grey area of the figure is the area for which the AUC is computed.

The experience, E Depending on the experience that they can have during the learning process, a machine learning algorithm can be supervised or unsupervised.

- In unsupervised learning, the algorithm learns properties of the structure of an experienced dataset consisting of a collection of features. An example of unsupervised learning algorithm is clustering. It aims at partitioning the dataset into clusters of examples with similar properties.
- In supervised learning, the algorithm experiences a dataset in which examples are associated with a label of target class. It then learns to predict the label y from $\mathbf{x}$

3.2.2 Capacity, Overfitting and Underfitting

In a machine learning system, the training stage consists of making the classification accuracy on the training set as high as possible. The system is then tested on the test set. Its accuracy will logically be smaller on the test set

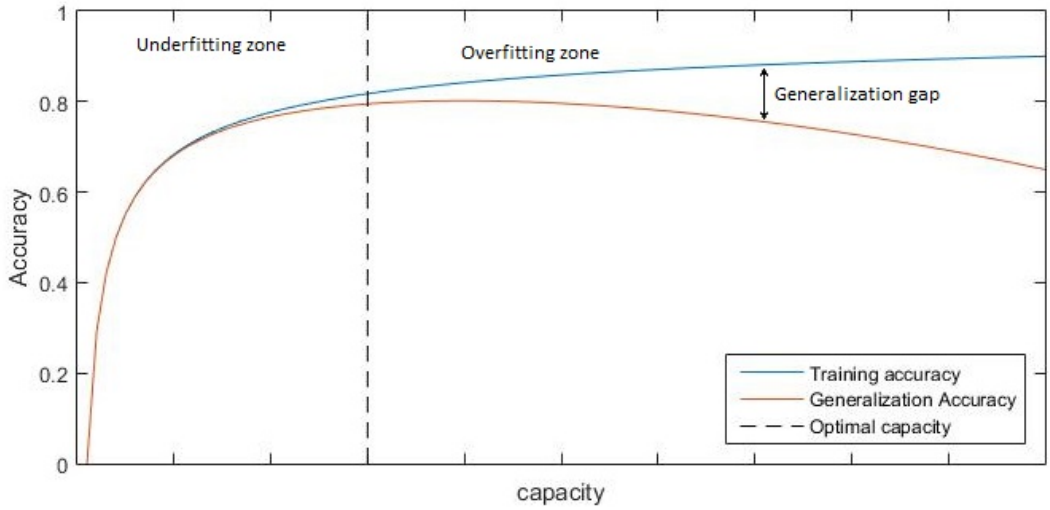


Figure 3.2: Relationship between capacity and accuracy. Training and generalisation accuracy have a different behaviour. At the left of the graph is the underfitting regime, where the training and test accuracy are low. When the capacity increases, the training accuracy increases but the generalisation accuracy starts decreasing. As a result, the gap between both accuracy increases. When the gap becomes too large, it is the overfitting regime where the capacity is above optimal capacity.

than the training accuracy. The better the system, the smaller the gap between those two accuracy is.

Underfitting happens when the model is unable to achieve a high accuracy on the training set. Overfitting is when the gap between training accuracy and generalisation accuracy is too large [36]. The model's capacity controls whether it is more likely to overfit or to underfit. The capacity can be defined as the model's ability to fit a large variety of functions. A low capacity results in underfitting as the model is not able to fit the training set. A high capacity can lead to overfitting as the model learns characteristics of the training set that are useless for generalisation.

The best performing algorithms need to have an appropriate capacity to the complexity of the task at hand. Figure 3.2 shows the typical relationship between capacity and accuracy.

Parametric and non parametric models. A parametric model learns a function described by a parameter vector with a finite size set ahead of any data observation. An example of parametric classification model is the logistic regression classifier. A non parametric model does not have this limitation and has a complexity that is a function of the training set size. The capacity of a non parametric model can therefore become very high. An example of non parametric model is the k-nearest neighbour classifier.

3.2.3 Hyperparameters and Validation Sets

In the majority of machine learning models, there is a number of parameters that control the algorithm's behaviour – by controlling its capacity for instance. These are called hyperparameters and are not tuned by the learning algorithm itself. They thus need optimising. For this purpose, a nested learning procedure must be implemented to be able to learn the hyperparameters resulting in the best system performance.

If the hyperparameter controls the capacity, it is not possible to learn it on the training set because as shown on Figure 3.2, the accuracy will increase with the capacity and the hyperparameters giving the highest accuracy will always be picked. This would result in overfitting and give poor generalisation. It is also important that the test examples are not used to make any decision about the model, including the choice of hyperparameters, as this would mean

that these example are not "unseen" by the model during testing. This would result in an optimistic evaluation of the model's generalisation accuracy. To solve this problem, a validation set is often used. It is a set of unseen examples by the training algorithm picked from the training data. The training data is therefore split into two subsets, one is used to learn the parameters of the learning algorithm, and the other one, the validation set, is used for selecting the hyperparameters. In this thesis, we used 80 % of the training data for training and 20 % for validation. The test set is then used for generalisation, once the hyperparameters have been optimised.

3.2.4 Cross-Validation

The datasets used in this thesis being small, a fixed training and test set would be problematic as estimating the average generalisation accuracy would imply statistical uncertainty. This would make it hard to compare different systems performances. In order to resolve this issue, a cross-validation procedure can be carried out: the data-generating process is repeated on different randomly chosen splits of the dataset. More specifically, a k-fold cross-validation can be computed. The dataset is split into k folds – or non overlapping subsets – and k different trials are run using different training and test sets. On the i^{th} trial, the i^{th} subset is used as test set and the remaining data is used for training. This procedure may also be used for validation.

3.2.5 Feature Extraction

As described previously, a learning algorithm requires some features taken from the objects – in our case, images – to classify. These features are usually

vectors describing certain characteristics of the image. The choice of features is critical to the performance of the algorithm. They are a representation of the input data and, for this purpose, have a descriptive function. They also need to have a high discriminative power in order to differentiate regions or patterns in the input image. Finally, their size need to be as reduced as possible in order to avoid the curse of dimensionality (see Section 4.3.1). The next section will go through feature extraction techniques.

3.3 Previous Work on Texture-Based CAD of Colorectal and Prostate Cancer

3.3.1 The Generic CADS

A great number of various techniques are available in the literature for automatic diagnosis of cancer using microscopic images of biopsies. Different approaches are taken by the authors. Some of them aim at detecting a cancerous region on a whole slide, others focus on grading the stage of the cancer and the rest work towards a classification of regions with homogeneous diagnosis [15, 4]. The majority of these approaches revolve around the same workflow – see Figure 3.3. First, an image preprocessing phase is used to remove irrelevant noise, segment key objects, regions, or features in the image, or standardise the intensity and the scale of the image to maintain unity in colour or grey level and magnification throughout the whole dataset. Afterwards, a feature extraction phase tries to capture the characteristic information for the problem at stake. It popularly resorts to colour, texture and morphological features. The next phase is the feature selection where the most discriminative features

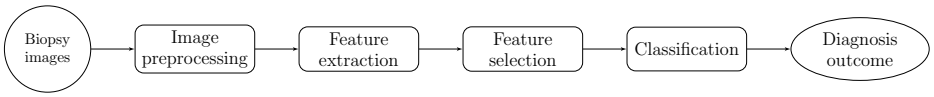


Figure 3.3: Standard workflow of a CAD algorithm

extracted from the image are picked out to be fed to the classifier. Finally, the classification phase either detects the cancerous region or gives a final diagnosis in terms of grade or type of tumour, depending on the authors' approach.

All these techniques can be divided into two different categories according to the feature extraction they use.

1. The algorithms using texture features: these algorithms use metrics to characterise the spatial variations of pixel intensity [37] in order to identify the representative patterns to the different diagnosis outcomes. Haralick *et al.* [38] indentified fine, coarse, smooth, rippled, molled, and irregular or lineated textures.

2. The algorithms using morphological features: this type of algorithm intends to estimate the shape and size of structures present on the image such as epithelial nuclei and cytoplasms, glands, lumen, mucous or certain types of cells like the goblet cells.

Using textures features has a clear advantage as cancerous and pre-cancerous tissues often do not have recognisable structures – as described in Section 2.2.3 – thus making a morphologic description impossible. A number of combinations of morphologic and texture features have been explored in precedent works. Tuceryan *et al.* [37] distinguished four different groups of texture analysis methods.

1. Statistical methods use the spacial distribution of grey values of the images. Common examples are co-occurrence matrices and autocorrelation features.

2. Topological methods characterise a texture as consisting of "texture elements" or primitives. Some methods then extract statistical properties of the primitives and use them as feature vectors. Instead, other methods look for a placement rule that constitutes the texture. Examples of such methods are Voronoi tessellations or Delaunay triangulations.

3. Model based methods are not only capable of describing the texture but they are also able to synthetise it thanks to parametres that capture the main perceived qualities of texture. Random fields models such as Markov random fields and fractals are the most common examples of such methods in literature.

4. Signal processing methods such as spacial domain filters, Fourier transforms, Gabor and wavelet models.

Table 3.1 summarises the different methods used in the context of CAD of colorectal and prostate cancer.

A complete CADS based on digitalised biopsy images of prostate or colorectum should include two steps. First, it should be able to locate the cancerous or abnormal region on a slide. Then, an automated and thorough analysis of this region determines the final diagnosis by finding the type of tumour or grading the cancer. However, in the majority of published works, authors focus on one of these two steps. This thesis concentrate on the second step.

Table 3.1: Summary of the different texture feature extraction methods

Method	Category	Description	Features
Statistical	Co-occurence Matrices (GLCM)	Captures spacial relationships between pixels with the same intensity	Haralick features [21, 39, 40, 41, 42, 43, 44, 45, 17, 18, 46, 47]
	Run-Length Matrices (GL-RLM)		GL-RLM features [48]
	Autocorrelation features	describes the coarseness, regularity and fineness of the texture	Autocorrelation coefficients
	Histogram-based	Compiles the different pixel intensities without any spacial information	First-order statistics [44, 43, 48, 17, 18], color-channel histogram [49, 50]
	Local operators	Captures the local information of texture	Local Binary Pattern (LBP) [51, 52, 53, 16],Local ternay Pattern (LTP)
Topological	Graphs	Represents the placement rule that depicts the texture	Statistics from graphs based on the nuclei positions [54, 50, 55]
Model-based	Random field models	Models the texture as a probability model or as a linear combination of a set of basic functions	Probabilistic pair-wise Markov model (PPMM) [56]
	Fractals	Uses scaling invariance to capture self-similarity of the image	Greyscale fractal dimension [57, 49] color fractal dimension [56, 58]
Signal processing-based	Time domain filter response	Captures information on orientation and edges of the image	Sobel, gradient and derivative [44, 43]
	Wavelets	Multiscale tool that captures both spacial and spectral information	Gabor filters [43, 44, 59], multiwavelets [40, 46, 51]

3.3.2 State-of-the-Art Texture-Based Tumour Classification and Grading for Digitalised Biopsy Images of Colon and Prostate Tumours

This section reviews the methods described in published articles for classification or grading of colon and prostate cancer using biopsy images. The main issue when comparing these studies lies in the diversity of data used by researchers testing their work. The datasets vary in size, ranging from a few dozen images to a few thousands. The number of classes used to distinguish two cases can also be very different from one study to the other. Some can use a binary cancer/not cancer distinction when others tend to display a more precise spectrum of cancer evolution and highlight the diversity of tumour types. The very type of data used can be either a panchromatic grey-scale image, or a RGB colour image, or even a multispectral image.

Feature extraction

Panchromatic and colour images Esgiar *et al.* [60] computed a gray-level co-occurrence matrix (GLCM) on each colorectal histological image and extracted some of the GLCM features proposed by Haralick [38]. In [47], Kalkan *at al.* combined the same features with structural ones before computing a feature selection and a four-class classification, achieving a 75.15 % accuracy. In the case of prostate cancer, several authors used the GLCM features [21, 43, 44, 41]. Haralick and morpologic features are sometimes combined. For instance, the authors of [41] used morphologic characteristics to classify non-cancerous regions by assuming that the lumen occupies a larger area of the image in a normal tissue. The classification between stroma and

cancerous tissue was then based on Haralick texture features. Their reported classification error was 20.7 %. In [51], the authors used a 8-class dataset of 5000 images. The classes involved were the following: tumor epithelium, simple stroma, complex stroma, immune cells, debris, normal mucosal glands adipose tissue and background (no tissue). The authors compared several texture descriptors such as GLCM, Linear Binary Pattern (LBP), perception feature – mimicking the human perception at an abstract level, as described in details in [61] – and Gabor filters. Their best reported result was achieved with a combination of LBP, GLCM, lower and higher order histogram features and perceptual features with 87.4 % accuracy. The authors of [57, 49, 58, 62] used fractal analysis for prostate cancer grading or carcinoma detection. Huang *et al.* [58] used two different fractal measurements: the conventional fractal dimension and an entropy-based fractal dimension. They achieved a 95 % accuracy for their system. In [49, 63], Tabesh *et al.* described the colour, the texture and morphologic characteristics of the tissue sample using object- and image-level features. Tissue structures are segmented and the intensity of these segmented regions are used for object-level features, while features such as colour channel histograms, fractal dimension and wavelet coefficient statistics are considered for image-level analysis. In this study, 96.7 % of the samples were correctly classified for the binary cancer versus non-cancer problem. When tackling the Gleason grading classification task, the accuracy was 81 %. Yu *et al.* [56] proposed a method using the colour fractal dimension that captures colour and textural information on the tissue. It is modeled as a mixture of gamma distributions per pixel. The spacial dependences between pixels are taken into account via a probabilistic pairwise Markov model (PPMM) [64, 65, 66] once a Bayesian classification between cancer and benign pixels has been performed. Jafari-Khouzani *et al.* [40] extracted the energy

and entropy of multiwavelet transform coefficients which they combined with Haralick features. These descriptors were used on a 100-image dataset split into 4 classes and produced a classification accuracy of 97 %. Almuntashri *et al.* [67], used a 3-class dataset of 45 images. They developed a system employing some of the wavelet transforms energy features along with wavelet-based fractal dimensions. This system reportedly performed a 95 % classification accuracy. In some studies, morphologic features are first extracted to be used as a basis for a texture feature extraction algorithm. This is the principle used by Naik *et al.* in [55] where cell nuclei were segmented and their centroids were used as seeds of Voronoi, Delaunay and minimum spanning tree graphs. Those graphs aim at capturing the nuclei's spacial organisation in the tissue to classify thanks to features like area, edge length and nuclear density. In [68], Sengar *et al.* used a mix of statistical an textural features extracted from preprocessed ROIs. Banwari *et al.*[69] used similar features for ROI segmentation. In [70] *et al.* Haralick and LBP features were extracted from preprocessed patches of the colon slide images converted from RGB to grey-level format. Similarly, Hussain *et al.* [**?**] compared the performances of different textural and morphological features coupled with different types of classifiers. They used a dataset of prostate cancer RGB images converted to the grey-level format. Other studies use a combination of textural and morphological features. For instance, Nguyen *et al.* [44, 71] segmented the nuclei using a maximum likelihood algorithm and combined it to a collection of texture features including first order statistics, Gabor filters statistics and Haralick features.

Multispectral images Multispectral images have been used for texture feature extraction. In [45], Masood *et al.* applied GLCM features after segment-

ing the image data through a pre-processing phase. the approach consists of using the spectral dimensions to segment the image into four clusters representing four different tissue types: nuclei, cytoplasm, glands and stroma. Chaddad *et al.* proposed an improved version of the snake algorithm for the segmentation and extraction of GLCM texture features of multispectral-segmented images [46]. In [72], the authors proposed a method for characterising the continuum of colorectal cancer using several texture features after segmentation. As for features extraction, the GLCM features, the Laplacian of Gaussian and discrete wavelets were used. A few other studies used wavelet transforms [44, 73] and Laplacian of Gaussian [47]. In [17], Roula *et al.* worked on prostate histological images and extracted GLCM features from each spectral band and combined them with morphological features for the discrimination phase using a quadratic discriminant analysis. They showed that multispectral analysis significantly improved classification scores. In [19], Tahir *et al.* first extracted statistical and structural features as well as the GLCM features. They then used a Round-Robin Tabu Search for dimensional reduction of the multispectral data before classification. They achieved a classification accuracy ranging between 98 % and 100 %.

None of the previously mentioned authors used a multispectral texture feature detector that uses the spectral dimension directly. They either combined several results of 2-dimensional texture detector run on each spectral band, or used dimensional reduction to create a 2D image on which the texture was to be detected. Khelifi *et al.* [74, 75, 76, 39] developed a multi-band texture detection extending the GLCM. For this purpose, they used a spatial and spectral grey-level dependence method assuming a joint information between spectral bands exists. They applied this technique to the prostate cancer case. This method was inspired by the generalised co-occurrence matrix presented

in [77] by Hauta-Kasari *et al.*. In [78], the authors extracted significant features of the GLCM of the images using a analysis of variance. They further reduced the feature dimensionality using PCA and finally trained a decision tree classifier, they achieved 92.59% accuracy.

However, only few studies use LBP texture features in this field [52, 53] and none of them uses the joint information of spatial and spectral dimensions. For example, the authors of [52] select a single band from which the texture extraction is conducted. In [53], the LBP histogram is built on all three colour channels of the image.

Multispectral images were also used in other cancer detections. Grote *et al.* [79] used multispectral images to distinguish between non-malignant lobular tissue from well differentiated breast cancer. They used a texture-based supervised classification in order to detect lobule candidate regions. Irshad *et al.*[80] worked on multispectral band selection applied to mitosis detection in breast cancer histopathology. They used texture features including Haralick and GL-RLM on selected bands before classification. Zimmerman-Moreno *et al.* [81] used LBP features extracted from each spectral band of lymph nodes microscopic images

Classification

After choosing a set of features to extract from the image, and potentially selecting a subset of them, the next step is the choice of a classification method. Like in most fields where machine learning is involved, the published studies on automatic diagnosis of colorectal or prostate cancer use a supervised learning approach. Machine learning systems use mathematical functions – called

classifiers – whose argument is a feature vector and returns one of the classes considered in the classification problem at hand. In supervised learning, this classifier requires a dataset to be trained on. During training, the classifier is fed a collection of pairs of feature vector and class label. It also establishes some decision boundaries in the feature space.

The most common classification methods used for this problem are k nearest neighbours (k-NN), support vector machines (SVM), neural networks, logistic classifier, random forest or linear discrimination. Alexandratou *et al.* [82] compared 16 supervised machine learning algorithms on how well they could classify a dataset of prostate cancer images according to either a tumour versus non-tumour split, a low- versus high-grade division, or a four-class diagnostic and grading problem. Thirteen Haralick features were extracted and WEKA (Waikato Environment for knowledge Analysis [83]) packages were used to evaluate the different classifiers. They concluded that Logistic Regression and SVM were the two most competitive classifiers.

Ensemble learning or multiclassifiers [84, 85] is a possible method used to improve a system's classification results. With this strategy, an ensemble – or collection – of classifiers is put together and each one of them is trained on all or part of the feature space. In the end, their predictions are combined to give the final outcome of the model. With a complex structure and multiple different characteristics and feature vectors, histologic images are particularity well suited for this type of strategy. For instance, by using the strengths of each classification method in discriminating some particular aspects of the task at hand and combining them, it can be expected that the overall performance of the system will be improved. Ensemble methods can work on different parts of the feature space, using different training sets or different classifiers all to-

gether. Doyle *et al.* [86, 87, 88] developed a cascaded ensemble learning system dividing the multiclass problem into several binary problems, going from the broadest to the most specific. Tissues are first sorted out between cancerous and non-cancerous. Then, the cancerous tissues are subdivided according to another binary classifier – e.g. Gleason grades 3 and 4 versus Gleason grade 5. This same approach is used until all the classes are processed. Through this process, the most different classes are better divided and it results in an increased accuracy. This method has proved to outperform the traditional one-versus-all scheme used for multiclass problems and the one-shot classification. The overall multiclass accuracy was 89 %. Nguyen *et al.* [44] used two SVM classifiers trained on different feature sets, meaning one of them is trained on texture features and the other is trained on morphological features. The probabilities that each classifier classifies their associated feature set as cancer or normal tissue are multiplied. Those products are then compared and the sample is classified as cancerous if the product of the probabilities that it is cancerous is greater than the product of the probabilities that it is normal tissue. Greenbalt *et al.* [53] presented a two-stage ensemble learning system that first assigns an initial grade using quaternion wavelets and LPB associated with a neural network multiclass classification. In a second phase, the classification result is refined using a SVM classifier if some classes have close probabilities. An accuracy of 98.9 % was reported over all the classes considered. Such a system can be generalised to more than two stages using a tree-like structure.

Sanghavi *et al.* [89], proposed a method based on scale-invariant feature transform (SIFT) and speeded up robust features (SURF) extraction on each colour plan, allowing them to extract key points on the image such as cell nuclei. This is followed by the creation of a dictionary of words using a k-means clustering

technique. The final classification is performed with a k-NN classifier. A 94 % classification accuracy was achieved for grade 3 and 4 of prostate cancer.

Table 3.2: Summary of the systems used for CAD of colorectal and prostate cancer

Author	**Feature**	**Classifier**	**Dataset**	**System performance**
Esgiar *et al.* [60]	Haralick features	Linear Discriminant Analysis (LDA) or k-NN	44 normal and 58 cancerous panchromatic images of colon biopsies	Accuracy: 90.2 %
Kalkan *et al.* [47]	Haralick features, morphologic features	Logistic regression classifier	55 panchromatic images of colon biopsy divided into 4 classes	AUC: 0.90-0.95
Kather *et al.* [51]	Haralick, LBP, lower and higer order statistics, perceptual features	SVM with radial basis function (rbf)	5000 panchromatic images of colon biposy divided into 8 classes	Accuracy 87.4 %
Huang *et al.* [58]	Fractal dimension, entropy based fractal dimension	Bayesian, k-NN, SVM	205 panchromatic images of prostate biopsy divided into 4 classes	Accuracy: 94.6 %
Tabesh *et al.* [63, 49]	Fractal dimension, color channel histograms, wavelet coefficient statistics	Bayesian, k-NN, SVM	2 sets of prostate panchromatic images: tumour/non-tumour 2-class set (367 images) and Gleason grade 4-class set (268 images)	Accuracy: Set 1: 96.7 % Set 2: 81.0 %
Yu *et al.* [56]	Colour fractal dimension and PPMM	Markov random field	27 panchromatic images of prostate biopsy	AUC: 0.831

Continued on next page

Table 3.2 – *Continued from previous page*

Author	**feature**	**Classifier**	**Dataset**	**System performance**
Naik *et al.* [55]	Graph-based features	SVM	44 panchromatic images of prostate biopsy divided into 3 classes	Accuracy: 91.48 %
Almuntashri *et al.* [67]	Wavelets energy features, wavelet-based fractal dimensions	SVM	45 panchromatic images divided into 3 classes	Accuracy: 95 %
Nguyen *et al.* [44]	morphologic features, Haralick features, first order statistics, Gabor filter statistics	SVM	17 panchromatic images of prostate biopsy	FPR: 6 %
Sun *et al.* [48]	Run-length matrix features	Multilayer Perceptron	9 panchromatic images of prostate biopsy	Accuracy: 89.5 %
Masood *et al.* [45]	Haralick features on segmented images using spectral dimensions	LDA, SVM	32 hyperspectral images of colon biopsy, 2 classes	Accuracy: 90 %
Roula *et al.* [17]	Haralick features from each spectral band, morphologic features	Quadratic discriminant analysis	33 Multispectral images of colon with 33 spectral bands between 400 nm and 720 nm	Error rate: 5.1 %
Tahir *et al.* [8]	Haralick features, Round-Robin Tabu Search	SVM	**prostate dataset**	Accuracy: 98 %

Table 3.2 summaries a selected number of methods presented in this section. The results are difficult to compare as each study uses different performance

measures as well as a different dataset with different data types and a different number of classes.

3.3.3 Previous Work on Multispectral Texture Analysis

Some methods for other applications, such as image segmentation, used a 3D histogram as a mean to fuse information from three colour channels of a colour image [90]. Hassan El Maia *et al.*[91] proposed a method for multispectral image classification using the mutual information of GLCM features. In [92], the authors used a method developed in [93] for automatic face recognition. This algorithm was a modified LBP that computed a LBP on each colour band of the spectrum separately and added opponent features to capture the spacial correlation between the bands. Radu-Mihai Coliban *et al.*[94] proposed a pseudo-morphology based on the Euclidean distance in $\mathbb{R}^n$. Using the proposed pseudo-morphology, the authors introduced a pseudo-granulometry and a morphological covariance to characterise the image texture. In [95], the authors use a neural network structure to classify multispectral texture information extracted from the images.

3.3.4 Previous Work on IR Analysis

In the field of facial recognition, the IR spectrum has been used and has proved to increase the recognition rates in many cases. Abdelhakim Bendada *et al.*[96] introduced a differential local ternary pattern (LTP) descriptor and extend their method to the IR spectrum. It was shown that a high recognition rate was achieved with the IR spectrum. The authors of [97] developed a method for synthesising the visible and near IR face images in order to take

advantage of both the illumination invariance of IR images and the detailed texture information provided by the face images captured in the visible range of the electromagnetic spectrum. The authors compared their method to the conventional LBP applied separately to the near IR and visible images and showed that the combined use of the IR and visible spectra increased the identification rate by 8.76 pp (from 88.83 % to 97.59 %). Thematic mapping imagery uses the infrared spectrum to acquire information that is not captured by the visible spectrum. Yun Zang [98] used an algorithm of conditional variance detection on multispectral images captured on a visible and IR spectrum for classification of urban treed areas.

Larsh *et al.* [22] worked on infrared spectroscopy of human cells and tissue for disease detection. The authors showed that it is possible to differentiate between IR spectra from the cytoplasm from those from the nucleus even in dividing cells. Smolina *et al.* [99] also demonstrated that the IR spectral response of epithelial tumours for breast cancers have a high discriminative power. In [100], IR imaging is used to predict the presence of regional or distant metastases in primary skin melanomas. Wolthuis *et al.* [101], used reconstructed colour-coded spectral images for generating an automatic IR-based histology of human colon carcinomas. The effectiveness of the IR spectral imaging for tumour heterogeneity characterisation and tissue subtype recognition was established. These previous research on IR imaging show that information that is invisible to the human visual system can be captured by IR imaging. This can lead to improved prevision performances for many problems. This type of imagery have not been directly applied to the problem at hand.

3.4 Conclusion

This chapter described the basics of machine learning algorithms and how texture features could be extracted in order to be used by the learning algorithm. The state-of-the-art texture-based computer aided-diagnosis systems for prostate and colorectal cancers were reviewed. We also addressed previous works on multispectral and IR texture analysis. The next chapter investigates different texture extraction methods and classifiers for multispectral histology images.

Chapter 4

Texture Analysis on Multispectral Images for Colorectal and Prostate Cancer Diagnosis

4.1 Introduction

A number of techniques have been used to characterise the texture of an image as discussed in Section 3.3.1. This chapter, based on the study [2], investigates four different types of texture feature extraction techniques: Haralick texture feature, Local Binary Pattern (LBP), a multiscale version of LBP and Local Intensity Order Pattern (LIOP) with the aim to evaluate their performances.

The proposed algorithm consists of extracting the features on each spectral band and fuse them by concatenating the features into a large feature vector.

Then, a Principal Components Analysis (PCA) is used in order to reduce the large dimensionality of the feature vector with a view to select the best features and hence avoid the effect of feature correlation which can negatively affect the classification accuracy.

To assess the usefulness of the proposed method, a comparative study against one similar algorithm using panchromatic images is carried out. The panchromatic image is obtained by averaging all the spectral bands into one two-dimensional image.

4.2 Texture Analysis

4.2.1 Haralick Texture Features

The Haralick features [38] are calculated from the grey-level co-occurrence matrix (GLCM) which reflects how often a pixel with the intensity value i occurs in a specific spatial relationship $(r, \theta) \in \mathbb{R}^2$ to a pixel with the value j. Namely, the GLCM of an $n \times m$ image with p different pixel values is a $p \times p$ matrix defined as follows [38]:

$$GLCM_{r,\theta}(i,j) = \sum_{x=1}^{n} \sum_{y=1}^{m} \begin{cases} 1, \text{ if } I(x,y) = i \text{ and } I(x + r\cos\theta, y + r\sin\theta) = j, \\ 0, \text{ otherwise.} \end{cases} \tag{4.1}$$

Four different spatial relationships are computed: $r = 1$ and $\theta = 0°, 45°, 90°, 135°$. This results in four different GLCMs: $GLCM_{1,0}$, $GLCM_{1,\frac{\pi}{4}}$, $GLCM_{1,\frac{\pi}{2}}$, $GLCM_{1,\frac{3\pi}{4}}$. These matrices are then normalised to have real values in $[0, 1]$.

The following Haralick features are computed from the normalised GLCM matrices $GLCM_{r,\theta}(i,j)$ of the image [38]:

- Energy:

$$\sum_{i,j} p(i,j)^2, \tag{4.2}$$

- Contrast:

$$\sum_{i,j} |i-j|^2 p(i,j), \tag{4.3}$$

- Homogeneity

$$\sum_{i,j} \frac{p(i,j)}{1+|i-j|}, \tag{4.4}$$

- Correlation

$$\sum_{i,j} \frac{(i-i)(j-j)}{\sigma_i \sigma_j} p(i,j). \tag{4.5}$$

A feature normalisation is then operated and these different Haralick features from all the GLCMs are concatenated in a vector. As a result, a vector of length 16 is created and used as a feature vector and image descriptor.

For multispectral images, GLCMs are extracted from each band and the Haralick features of each band are computed separately. They are then concatenated into a single final vector used as image descriptor.

4.2.2 Local Binary Pattern (LBP)

Ojala *et al.* [102] described LBP texture features as a local characterisation of a pixel neighbourhood at a radius R sampled into a set of P neighbors on a circle centred around the central pixel and of radius R. Let g_0 be the intensity

of the central pixel x and g_p the intensity of its p^{th} neighbor. The LBP is defined as follows [102]:

$$LBP_{P,R}(x) = \sum_{p=1}^{P} s(g_0 - g_p)2^{p-1}, \tag{4.6}$$

where,

$$(x) = \begin{cases} 0, \text{ if x} \leq 0, \\ 1, \text{ if x} > 0. \end{cases} \tag{4.7}$$

LBP is computed for the whole image, before it is pooled into a LBP histogram of size 256. The resulting LBP histogram, which is invariant to intensity changes, is then used as a texture feature descriptor to characterise the image.

A multiscale version of LBP has also been modified and tested on the images: the LBP histograms are calculated over different scales and concatenated into a single multiscale LBP histogram. For each scale, the neighbourhood is considered at a different radius R.

4.2.3 Local Intensity Order Pattern (LIOP)

The global ordinal intensity information is used to divide the image into subregions where local ordinal information of each pixel is accumulated into their respective LIOPs [103]. More precisely, the first phase consists of a preprocessing step where a Gaussian filter is applied in order to smooth an image, therefore making the relative order insensitive to noise. Secondly, the pixels are sorted by their intensity in a non-descending order. A histogram is created by dividing this array of pixels into B equal bins, each bin representing a subregion.

Table 4.1: Index Table of the permutations in $(1, 2, 3)$

Permutation	(1,2,3)	(1,3,2)	(2,1,3)	(2,3,1)	(3,1,2)	(3,2,1)
Index	1	2	3	4	5	6

Then, the LIOP is computed for every subregion. Each pixel's, x, neighbourhood is sampled into a set of N neighbours $(x_1, x_2, \ldots, x_N)$. Let's consider Π^N the set of all possible permutations π of N integers $(1, 2, \ldots, N)$ and set an index table defining a function $Ind(\pi)$ as shown is Table 4.1 for $N = 3$.

The set of neighbours is sorted in an intensity non-descending order to obtain a permutation π of the original set. The $LIOP$ for pixel x is a $N!$-dimensional vector defined as follows:

$$LIOP(x) = (0, 0, \ldots, \underset{Ind(\pi)}{1}, \ldots, 0). \tag{4.8}$$

For each subregion, a $N!$ bins histogram is created with the LIOPs of all the pixels within it. They are then concatenated to form the LIOP descriptor of the image.

$$LIOP_{descriptor} = (des_1, des_2, , des_B), \tag{4.9}$$

where:

$$des_i = \sum_{x \in bin_i} LIOP(x). \tag{4.10}$$

This method captures both global and local intensity information and makes the features invariant to intensity changes and geometrical and photometric transformations such as rotation.

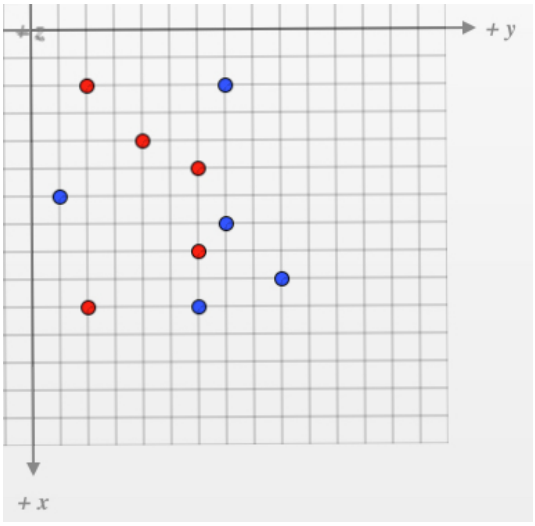
4.3 Feature Selection

Prior to performing the sample classification, a feature selection is operated. The approach used to handle multispectral data generates large feature vectors. Indeed, texture feature vectors are extracted from each spectral band of an image and are then concatenated to form the image descriptor.

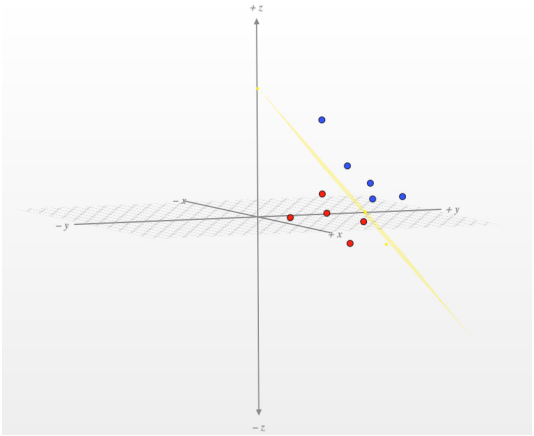
This can result in an increased training time, but can also lead to poorer results due to the curse of dimensionality problem causing an increased overfitting. This curse of dimensionality is especially visible with the dataset used in this work because they consist of a small amount of samples.

4.3.1 Curse of Dimensionality

The curse of dimensionality is a counter intuitive concept that states that, with a fixed number of training samples, the model's accuracy decreases when the number of explanatory variables increases [104, 105]. This can be explained by the fact that when the dimensionality of the feature space increases, it becomes sparser and sparser as the density of samples decreases. As a result, the likelihood that a sample lies on the wrong side of the best separating hyperplane tends to zero when the dimensionality tends to infinity. In other words, as illustrated in the example in Figure 4.1a and 4.1b, the probability of finding a hyperplane that separates correctly two classes in the training set increases with the dimensionality of the feature space. However, when the function is projected back into a lower dimensional feature space, the simple hyperplane becomes a complicated function. Consequently, this function learns characteristics that are specific to the training set, leading to overfitting. Therefore, the model fails to provide accurate results on the testing set.



(a) In a 2D feature space, this set of samples is not separable by a simple linear function



(b) By increasing the feature space dimensionality, it is possible to find a hyperplane that separates the samples

Figure 4.1: Example of a 2-class training set described in (a) a 2D feature space and (b) a 3D feature space

4.3.2 Principal Component Analysis

PCA is a widely used technique for feature selection [106, 107]. Its goal is to find a matrix $\mathbf{W}$ verifying eq. 4.11 so that the n-dimensional transform $\mathbf{s} = (s_1, s_2, \ldots, s_n)^T$ explains the maximum amount of variance using n linearly transformed orthogonal components $s_1, s_2, \ldots, s_n$.

$$\mathbf{s} = \mathbf{W}\mathbf{x}, \tag{4.11}$$

where, $\mathbf{x}$ is a m-dimensional random vector.

The PCA is computed using a recursive process. The direction of the first principal component, $\mathbf{w_1}$, is defined as follows:

$$\mathbf{w_1} = \arg \max_{||\mathbf{w}||=1} E\{(\mathbf{w}^T\mathbf{x})^2\}, \tag{4.12}$$

where, $\mathbf{w_1}$ is a m-dimensional vector. As a result, the first principal component, $\mathbf{w_1}$, is the projection in the direction for which the projection has the maximum variance.

The general term of the recursive formulation is defined as follows: having determined the $k-1$ first principal components, the k^{th} principal component is defined as the principal component of the residual:

$$\mathbf{w_k} = \arg \max_{||w||=k} E\{(\mathbf{w}^T(\mathbf{x} - \sum_{i=1}^{k-1}(\mathbf{w_i}\mathbf{w_i}^T\mathbf{x}))^2\}. \tag{4.13}$$

The principal components are given by $s_k = \mathbf{w_k}^T\mathbf{x}$. In practice, the PCA is usually carried out using the covariance matrix of the sample $\mathbf{C} = E\{\mathbf{x}\mathbf{x}^T\}$. The direction vectors, $\mathbf{w_k}$, of the principal components, s_k, are the eigenvectors

of $\mathbf{C}$ corresponding to its n largest eigenvalues. In this case, the goal is to perform a dimensionality reduction, therefore $n << m$ is chosen.

4.4 Classification

Once the texture features are extracted and the most discriminative selected, the classification is carried out. To achieve this step, the database is divided into a training set and a testing set, consisting in 70 % and 30 % of the total dataset, respectively. A 10-fold cross-validation is then computed and all the loops' results are averaged in order to obtain the final results.

As a measure of performance and to take into account false alarm rates, the classifier accuracy and the ROC's Area Under Curve (AUC) for each of the three classes are used.

Five different classifiers were tested to compare their performances: the k-Nearest Neighbour (k-NN) classifier, the Logistic Regression (LR) classifier, the Decision-Tree (DT) Classifier, the Random Forest (RF) used with 100 decision trees, and the Support Vector Machine (SVM) with a Gaussian kernel.

4.4.1 k-Nearest Neighbour (k-NN) Classifer

The k-NN classifier is a non-parametric method, i.e. it needs to store the training examples. In fact, in this method, a sample is classified by a majority voting scheme from its k closest training examples. The parameter k is the only parameter to tune in order to optimise accuracy on the testing set. A distance metric also needs to be selected. The more commonly used is the

Minkowski distance L^p:

$$L^p(\mathbf{x_i}, \mathbf{x_j}) = (\sum_k |x_{i,k} - x_{j,k}|^p)^{1/p}. \tag{4.14}$$

This distance is the Euclidean distance with $p = 2$. However depending on the data structure, other distances can be used such as Chi Square or cosine distances. For this work, the Euclidean distance was selected for its versatility and ease of implementation.

4.4.2 Logistic Regression (LR) Classifier

The LR classifier is easy to implement and is computationally inexpensive [36]. Once the training is performed, the classification step is also simple and rapid.

The principle of a LR classifier is simple: for every training example, a linear combination of its features is fed to a sigmoid function $\sigma(z)$:

$$\sigma(z) = \frac{1}{1 + e^{-z}}. \tag{4.15}$$

The sigmoid function is represented in Figure 4.2. It has a noticeably similar behaviour to a step function. However, the sigmoid function has continuity and differentiability properties on its whole definition domain.

As a result, the linear combination of features fed to the sigmoid function will return a value between 0 and 1. If this value is greater than 0.5, the example is classified as 1 otherwise, it is classified as 0.

More formally, this can be described as follows. Let the labeled data be $\{\mathbf{x_i}, y_i\}, i = 1 \dots n, y_i \in \{0, 1\}, \mathbf{x_i} \in \mathbb{R}^d$. The aim is to find a weight vector

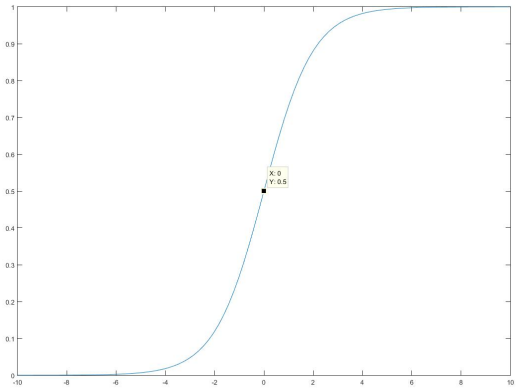


Figure 4.2: Plot of the sigmoid function. Its value at 0 is 0.5, and quickly tends to 1 when z increases while quickly vanishing when z decreases.

$\mathbf{w}$ so that for a maximum of samples $\mathbf{x_i}$, the following condition is respected:

$$\begin{cases} \sigma(\mathbf{x_i} \cdot \mathbf{w}) > 0.5 \text{ if } y_i = 1, \\ \sigma(\mathbf{x_i} \cdot \mathbf{w}) \leq 0.5 \text{ if } y_i = 0. \end{cases} \tag{4.16}$$

This is therefore an optimisation problem which is solved using a gradient descent method.

For the testing phase, the linear combination of the new example's features is fed to the sigmoid function, using the weights calculated on the training data.

4.4.3 Decision Tree (DT) Classifier

The basic principle of the DT classification model is to break down a complex decision making problem into a set of simpler decisions to make [108]. This often makes the solution easier to interpret. As illustrated in Figure 4.3, a DT is a tree in which each internal node is labeled with an input feature. The leaves of the tree are labeled with a class. During training, the problem of

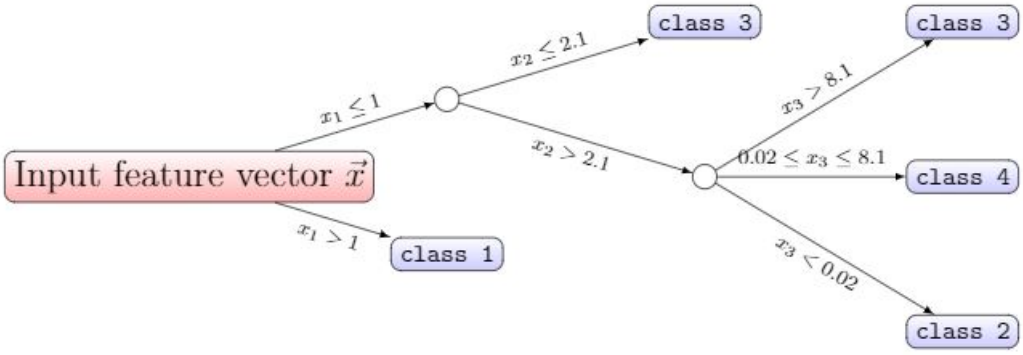


Figure 4.3: Diagram of an example of a DT classifier

classifying the full input feature vector is divided into smaller problems at the nodes. The deeper the tree is, the more layers it will have and the smaller the decision making problems will be at the internal nodes.

4.4.4 Random Forest (RF) Classifier

The RF classification method is based on an aggregation of DTs [109]. A random feature selection is realised and different sets of features are used to train different DTs. The different predictions of the DTs are submitted to a majority vote or an averaging principle to give the final prediction. An example of diagram of the RF algorithm is shown on Figure 4.4.

4.4.5 Support Vector Machine (SVM)

The SVM framework is a very popular approach for supervised learning. It has thee main properties that make it attractive for this purpose. [110]

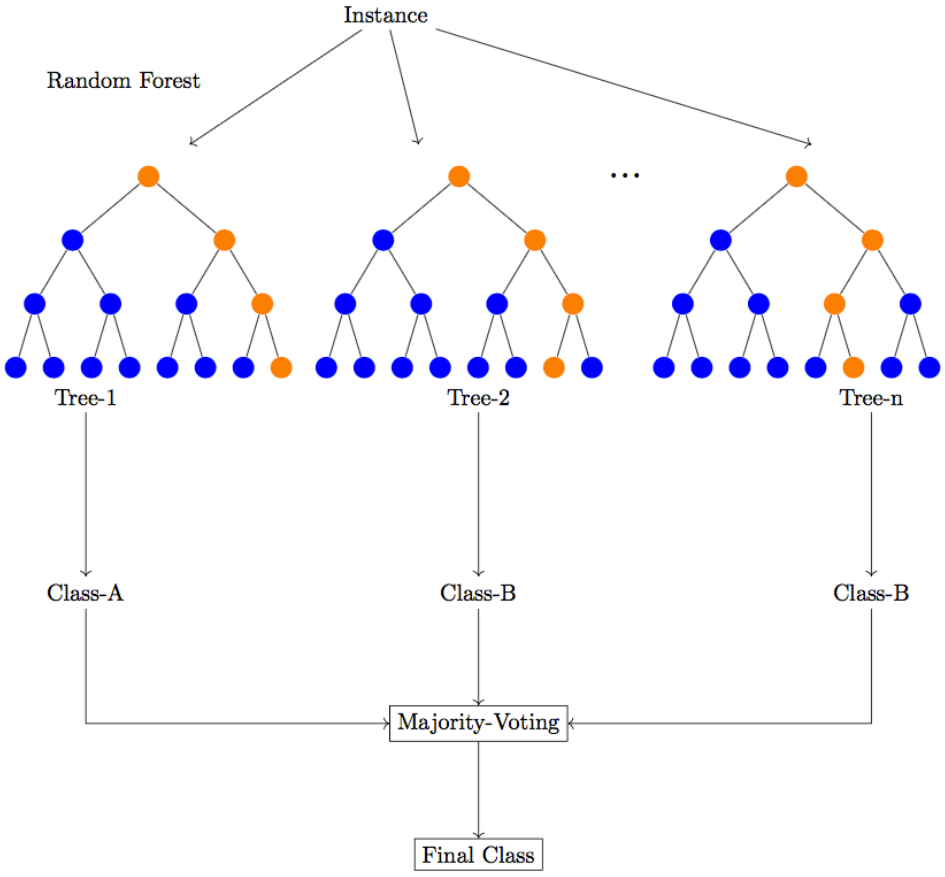


Figure 4.4: Diagram of the RF classifier

- SVMs build a maximum margin decision boundary, i.e. with the maximum distance to the samples. This helps them perform well on the generalisation set [85].

- The decision boundary is a linear hyperplane. However, by using the kernel trick, data can be embedded into a higher dimensional space. This trick is used for data that are not linearly separable in the original space. As seen in Section 4.3.1, increasing the feature space dimensionality often helps to find a linear separator. This separator is non-linear in the original feature space and therefore can separate non-linearly separable data.

- SVMs are a non-parametric method meaning that the training data are not summarised by a fixed set of parameter, i.e. the number of parameter depends on the number of training examples. However, in practice, they retain only a small number of training samples, usually proportional to the number of dimensions. Consequently, SVMs combine the advantages of both the non-parametric and parametric methods: they are able to depict complex functions whilst being resistant to overfitting.

The main idea of SVMs is that some training examples are more important than others: those from one class that are closer to the other class in the feature space. The decision boundary should therefore be the farthest away possible from the training examples, namely maximising the margin which is the distance between the closest example from each class and the decision boundary. Those closest training samples to the decision boundary are the support vectors. This principle is illustrated in Figure 4.5.

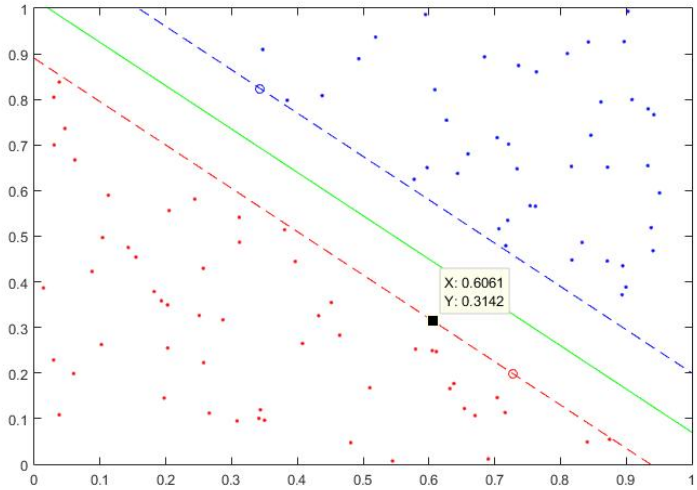


Figure 4.5: Example of a SVM binary classification. Red points correspond to the positive class and the blue to the negative class. The hard green line is the maximum margin separator and the margin is the area between the dashed lines. The support vectors are the points that are on the dashed lines, represented by a small circle marker.

Linear SVM

The data are assumed linearly separable by a separating hyperplane dividing the positive class from the negative one [111]. The data are labeled $\{\mathbf{x_i}, y_i\}$, $i = 1, \ldots, l$, $y \in \{-1, 1\}$, $\mathbf{x_i} \in \mathbb{R}^d$. The points $\mathbf{x}$ that are on the hyperplane satisfy Eq. 4.17.

$$\mathbf{w} \cdot \mathbf{x} + b = 0, \tag{4.17}$$

where, $\mathbf{w}$ is a normal vector to the hyperplane. Therefore, the distance between the hyperplane and the origin is given by $d_h = \frac{|b|}{||\mathbf{w}||}$, where $||\mathbf{w}||$ is the Euclidean norm of $\mathbf{w}$. Let d_+ and d_- be the distance from the separating plane and the closest positive and negative examples, respectively. The margin of the separating hyperplane equals $d_+ + d_-$. The aim of the SVM is to find the separating hyperplane that maximises this margin. This problem can be formulated as follows.

It can be assumed that all the training examples satisfy:

$$\mathbf{x_i} \cdot \mathbf{w} + b \leq -1 \text{ for } y_i = -1, \tag{4.18}$$

$$\mathbf{x_i} \cdot \mathbf{w} + b \geq +1 \text{ for } y_i = +1. \tag{4.19}$$

The following can be found from Eq. 4.18 and 4.19:

$$\forall i, \qquad y_i(\mathbf{x_i} \cdot \mathbf{w} + b) - 1 \geq 0. \tag{4.20}$$

The points for which the equality holds in Eq. 4.18 and 4.19 are on the hyperplanes $H_1 : \mathbf{x_i} \cdot \mathbf{w} + b = 1$ and $H_2 : \mathbf{x_i} \cdot \mathbf{w} + b = -1$, respectively, both with a normal vector $\mathbf{w}$. Therefore, the margin equals $d_+ + d_- = \frac{2}{||\mathbf{w}||}$. Consequently, finding the planes H_1 and H_2 that maximise the margin is equivalent to minimising $||w||^2$ under the constraints given by Eq. 4.20.

The problem is further solved by using a Lagrange multipliers method. Let $\alpha_i, i = 1, \ldots, l$, be the Lagrange multipliers for each inequality constraints of Eq. 4.20. The resulting Lagrangian is [111]:

$$L_P = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{l} \alpha_i y_i(\mathbf{x_i} \cdot \mathbf{w} + b) + \sum_{i=1}^{l} \alpha_i. \tag{4.21}$$

This Lagrangian L_P must now be minimised with respect to $\mathbf{w}$ and b, and in the meantime the derivative of L_P with respect to all the α_i must vanish with the constraints that $\alpha_i \geq 0$. The objective function being convex, and the points that satisfy the constraints forming a convex set, this problem is convex. As a result, it is equivalent to the dual problem [111, 112]: maximise

L_P subject to the constraints that its gradiant with respect to $\mathbf{w}$, and b vanish and that $\alpha_i \geq 0$.

The gradient of L_P vanishes if and only if:

$$\mathbf{w} = \sum_i \alpha_i y_i \mathbf{x_i}, \tag{4.22}$$

and

$$\sum_i \alpha_i y_i = 0. \tag{4.23}$$

By substituting them into Eq. 4.21, one obtains the final dual formulation of the problem:

$$L_D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j (\mathbf{x_i} \cdot \mathbf{x_j}). \tag{4.24}$$

The SVM training therefore consists in maximising L_D with respect to the α_i subject to constraints given by Eq. 4.23 and $\alpha_i \geq 0$. The problem solution is given by Eq. 4.22.

One should note that there is one α_i per training sample and those with $\alpha_i > 0$ are the support vectors, as they lie one of the hyperplanes H_1 or H_2. All the other points have $\alpha_i = 0$ and do not have an effect on the decision boundary.

Non-Linear SVM: Kernelisation

In the case of non linearly separable data, it is possible to find a linear separator in a higher dimensional feature space $\mathcal{H}$ by mapping the data from the original feature space $\mathcal{L}$ to $\mathcal{H}$ with a mapping function $F : \mathcal{L} \mapsto \mathcal{H}$ [111, 85].

In the linear case (Section 4.4.5), it can be observed that the data only appear in the form of a dot product, $\mathbf{x_i} \cdot \mathbf{x_j}$, in Eq. 4.24. Therefore, with this

transformation, the SVM training will be the same as in Section 4.4.5, except that the dot product in $\mathcal{L}$, $\mathbf{x_i} \cdot \mathbf{x_j}$, is replaced by the dot product in $\mathcal{H}$, that is: $F(\mathbf{x_i}) \cdot F(\mathbf{x_j})$.

This is where the kernel trick is accomplished. By defining a kernel function such that $K(\mathbf{x_i}, \mathbf{x_j}) = F(\mathbf{x_i}) \cdot F(\mathbf{x_j})$, the only knowledge of $K : \mathcal{L}^2 \mapsto \mathcal{H}$ is enough to perform the SVM training without explicitly knowing the mapping function F. Mercer's theorem states that any kernel function such that the matrix $\mathbf{K}_{ij} = K(\mathbf{x_i}, \mathbf{x_j})$ is positive definite, correspond to a feature space [113, 114]. Consequently, a reasonable kernel function can be chosen, and from its use, an optimal linear separator can be found efficiently in high-dimensional feature space. When mapped back into the original feature space, the decision boundary is non-linear.

In this thesis, SVM is used with the widely employed Gaussian kernel, given by:

$$K(\mathbf{x_i}, \mathbf{x_j}) = e^{-\frac{||\mathbf{x_i} - \mathbf{x_j}||^2}{2\gamma^2}}. \tag{4.25}$$

Two parameters are to be tuned depending on the input data: parameter C and γ. The parameter γ translates the radius of influence of the support vectors. A high γ means that the support vector will have a small radius of influence while a high value means that the influence of the support vectors reaches farther away [111]. The C parameter is a trade-off parameter. The lower the C value, the smoother the decision boundary is. However, a high value of C means the classification of all training examples needs to be accurate even if it means that the model selects more support vectors and has a more complex decision surface [111].

4.4.6 Multiclass Classification

For some classifiers, the multiclass algorithm can be obtained by extending the binary classifier. These include the random forest, neural network, SVM [115], k-Nearest Neighbour, and Naive Bayes.

However, many classifiers implementations - such as the widely used implementation of SVM, LibSVM [116] - commonly decompose the multiclass problem into several binary problems [111, 117]. It was even shown that these methods performed better than the extended versions of the classifiers in some cases [118, 119]. The two mainly used approaches are described here.

One-versus-All (OvsA) or One-versus-Rest

The idea of the OvsA approach is to decompose the K-class classification problem into K different binary problems, where each separates a class from the other $K-1$ classes [120]. Therefore, K binary classifiers are required, where the k^{th} classifier is trained with the samples from class k as positive examples and the samples belonging to the other $K-1$ classes as negative examples. During the testing phase, the classifier returning the maximum output wins and the label corresponding to its class is assigned to the sample tested.

One-versus-One (OvsO)

In the OvsO approach, classes are compared pairwise by different binary classifiers [121, 119]. For each pair of classes, only the samples from the two considered classes are used to train a binary classifier. This technique results

in constructing $\frac{K(K-1)}{2}$ different binary classifiers. During testing phase, a voting approach among the classifiers is taken. The class that was chosen by the higher number of classifiers is the winner and the label corresponding to its class is assigned to the sample tested.

Chosen Approach

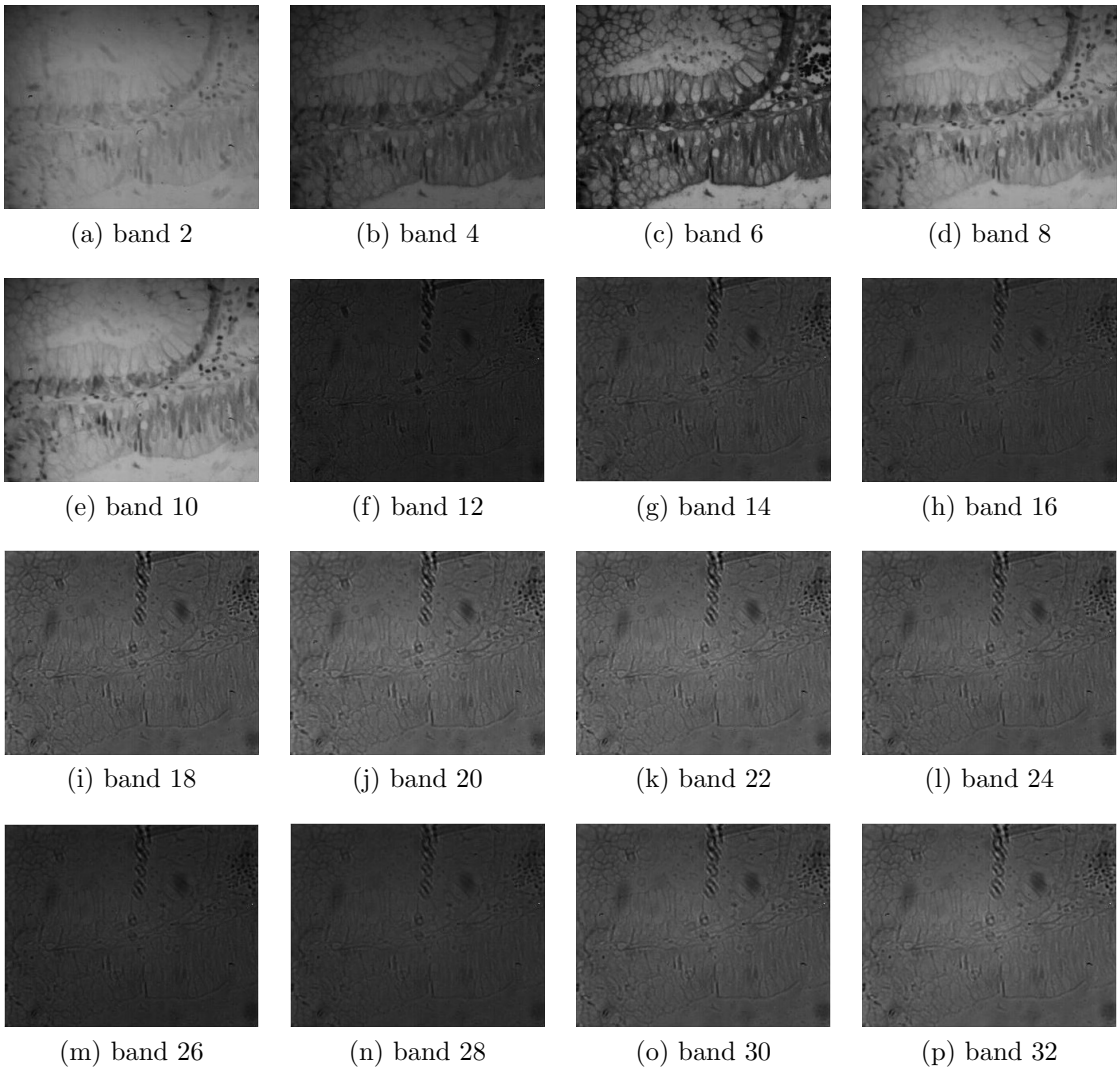
In this chapter and for ease of results interpretation, a OvsA method was used to generate multi class versions of the classifiers.

4.5 Experiments

Two sets of experiments were carried out. The first one was performed on panchromatic images that were obtained by averaging the multispectral images over their spectral bands in order to generate a single 2D image containing the contribution of all the spectral bands. Figure 4.7 shows an example of a panchromatic image, computed from the averaged spectral bands of the multispectral image displayed in Figure 4.6. This is a simulation of what a greyscale image would be. The second set of experiments was carried out using the multispectral images from the datasets. This comparison allows to assess the usefulness of multispectral imaging and the gain of information it causes.

4.5.1 Feature Extraction

In order to compare their performances, the texture features described in this chapter were extracted from the images of the two datasets detailed in Sec-



(a) band 2 (b) band 4 (c) band 6 (d) band 8

(e) band 10 (f) band 12 (g) band 14 (h) band 16

(i) band 18 (j) band 20 (k) band 22 (l) band 24

(m) band 26 (n) band 28 (o) band 30 (p) band 32

Figure 4.6: An extract from the spectral bands of a sample taken from the colorectal dataset

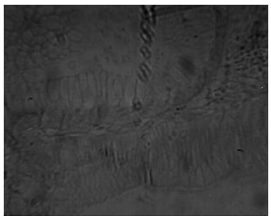


Figure 4.7: Resulting panchromatic image, averaged spectral bands

Table 4.2: Feature vector size

Feature type	Feature vector size		
	panchromatic images	Colorectal Dataset	Prostate Dataset
LIOP	144	6048	2304
LBP	256	10752	4096
Multiscale LBP	2048	86016	32768
Haralick	16	672	144

tion 2.6. Table 4.2 summarises the size of each feature vector for the panchromatic images. For multispectral images, this size is multiplied by the number of spectral bands and the resulting vector lengths are displayed in the table.

4.5.2 Feature Selection

PCA was conducted on the extracted features before classification. The optimal number of principal components was computed for each feature type. This was done by computing the cumulative sum of the explained variance ratio, i.e. the variance ratio for which the selected principal components account for. The number of principal components that explained 99 % of the variance was selected as the optimal number of principal components. This allows to be sure that the components used for classification contain nearly all the information from the features. The remaining information can be considered as noise and is filtered out. The lower number of components improves the computing time and avoids the curse of dimensionality. Table 4.3 shows the number of principal components selected for each type of feature for each dataset.

Table 4.3: Number of principal components selected

Feature type	colorectal		Prostate	
	Panchromatic	Multispectral	Panchromatic	Multispectral
LIOP	99	136	125	350
LBP	38	30	13	20
Multiscale LBP	112	31	29	135
Haralick	5	30	6	20

Table 4.4: Parameters C and γ of the SVM classifier

Feature type	colorectal		Prostate	
	Panchromatic	Multispectral	Panchromatic	Multispectral
LIOP	C: 1 γ: 10	C: 10 γ: 0.1	C: 1 γ: 10	C: 100 γ: 0.1
LBP	C: 10000 γ: 1	C: 1000 γ: 1	C: 100 γ: 10	C: 100 γ: 1
Multiscale LBP	C: 10 γ: 0.1	C: 10000 γ: 0.01	C: 100 γ: 0.1	C: 10 γ: 1
Haralick	C: 10000 γ: 0.0001	C: 100000 γ: 0.00001	C: 1 γ: 10	C: 100 γ: 0.1

4.5.3 Classification

The five classifiers described in Section 4.4 were used for the classification in order to compare their performances. For SVM and LR, a grid-search was performed in order to find their optimal parameters. Table 4.4 shows the parameters C and γ that were chosen with a grid search for the SVM classifier. Table 4.5 displays the parameter C of the LR classifiers. For the k-NN classifier, the rule introduced by Duda et al. [122], stating that the parameter k should be equal to the square root of the number of samples, was used. For the RF classifier, 300 trees were used. A 10-fold cross-validation scheme is used and the final results are averaged over the different runs of the cross-validation.

Table 4.5: Parameter C of the LR Classifier

Feature type	colorectal		Prostate	
	Panchromatic	Multispectral	Panchromatic	Multispectral
LIOP	0.01	10	1	1
LBP	10	1000	100	100000
Multiscale LBP	100	1000	100	1000000
Haralick	10	100	100	1000

4.6 Results and Analysis

Table 4.6 shows the accuracy obtained for the different combinations of texture features and classifier both for panchromatic and multispectral images. The highest accuracy is displayed in bold text for each dataset.

The table shows that, for every dataset, the best classification accuracy is achieved with the combination of multiscale LBP and SVM classifier. Kalkan et al. [47] used Haralick features with a LR classifier and tested their algorithm on panchromatic images (90 % accuracy). Kather et al. [51] used a combination of texture features with a SVM classifier on panchromatic images (87.4 % accuracy). Consequently, this makes the proposed algorithm similar when using panchromatic data and results show that the performances are also similar for this type of data. However, the datasets being different, it is not possible to directly compare the performance measures.

For both the colorectal and prostate datasets, the best accuracy was found with multispectral data. This shows how much more discriminative information can be extracted from the samples when they have been acquired with a multispectral imagery system.

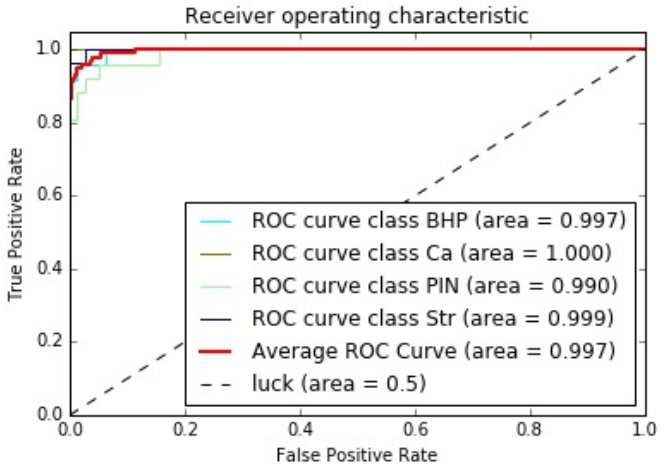
The accuracy is higher on the prostate dataset than it is on the colorectal one. This can be explained by a difference in size of the image. The prostate

Table 4.6: Performance of the different combinations of texture feature and classifier. The performance measure used here is the accuracy (in %) and its standard deviation is given in brackets

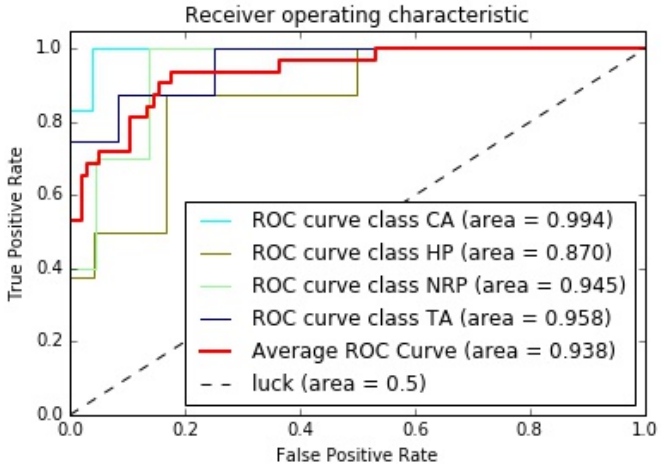
Feature	Classifier	colorectal		Prostate	
		Panchromatic	Multispectral	Panchromatic	Multispectral
LIOP	k-NN	33.2 (±1.9)	39.1 (±1.8)	41.0 (±1.7)	48.5 (±1.9)
	LR	40.4 (±0.8)	53.7 (±0.7)	52.8 (±1.0)	65.1 (±0.7)
	SVM	43.6 (±0.6)	55.1 (±0.5)	55.7 (±0.5)	68.0 (±0.5)
	DT	34.1 (±2.1)	31.0 (±1.9)	41.4 (±1.7)	45.9 (±1.8)
	RF	39.6 (±1.4)	39.5 (±1.3)	54.2 (±1.1)	57.8 (±1.1)
LBP	k-NN	30.2 (±1.3)	63.8 (±0.9)	70.8 (±0.9)	77.4 (±0.9)
	LR	40.0 (±1.4)	69.5 (±1.0)	76.8 (±0.8)	90.5 (±0.7)
	SVM	46.8 (±1.3)	77.6 (±0.8)	79.3 (±0.7)	91.6 (±0.9)
	DT	32.3 (±2.1)	64.4 (±1.6)	65.7 (±1.6)	79.0 (±1.3)
	RF	38.7 (±1.4)	80.7 (±1.3)	72.9 (±1.0)	81.1 (±1.2)
M. LBP	k-NN	31.5 (±1.2)	64.7 (±1.5)	75.3 (±1.0)	89.1 (±0.9)
	LR	51.1 (±1.3)	87.3 (±1.1)	84.9 (±0.7)	91.5 (±0.7)
	SVM	**51.3 (±0.7)**	**88.2 (±0.5)**	**88.9 (±0.6)**	**92.4 (±0.4)**
	DT	27.2 (±1.4)	68.5 (±1.1)	66.8 (±0.7)	79.5 (±0.7)
	RF	40.2 (±1.4)	82.8 (±1.3)	77.3 (±1.1)	88.0 (±1.0)
Haralick	k-NN	36.2 (±0.9)	55.6 (±1.0)	83.9 (±1.3)	87.2 (±0.9)
	LR	42.1 (±0.9)	86.5 (±1.0)	80.4 (±0.8)	89.1 (±0.9)
	SVM	41.7 (±0.9)	83.4 (±1.0)	85.1 (±0.7)	91.9 (±0.8)
	DT	29.3 (±2.8)	63.3 (±1.3)	77.2 (±1.6)	83.8 (±1.9)
	RF	33.9 (±1.0)	82.2 (±1.1)	83.1 (±1.4)	89.4 (±0.9)

dataset images being larger, it is easier for the system to learn texture patterns. Another reason for this difference of performance might come from the image quality.

Figure 4.8 and 4.9 shows that the ROC curves for Multispectral LBP features and SVM. For both datasets, the AUC is higher for class Ca than the others. This class representing the cancerous case, it has more specific characteristics than the other classes. Its texture in particular is chaotic compared to the more structured tissues present in the other classes. Consequently, it shows that the system performs better on the binary task of discriminating cancerous versus non-cancerous tissue than the other Ovs binary classifications.

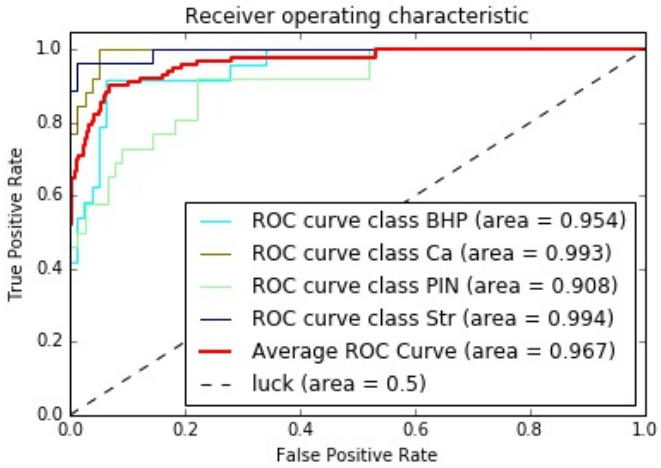


(a) ROC curve for the prostate dataset with multispectral data

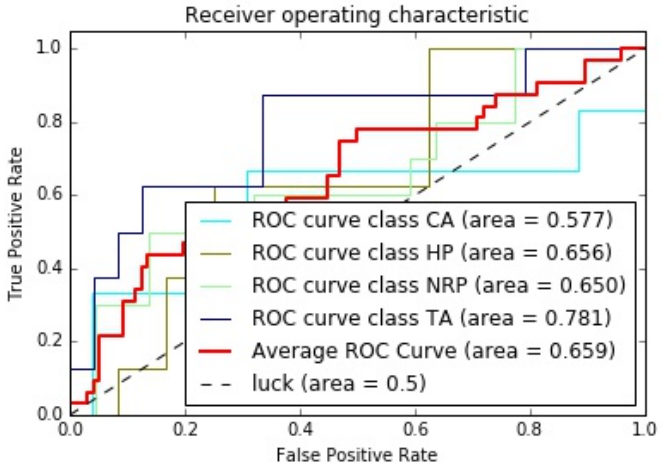


(b) ROC curve for the colorectal dataset with multispectral data

Figure 4.8: ROC curves for the Multiscale LBP and SVM combination with multispectral images



(a) ROC curve for the prostate dataset with panchromatic data



(b) ROC curve for the colorectal dataset with panchromatic data

Figure 4.9: ROC curves for the Multiscale LBP and SVM combination with panchromatic images

4.7 Conclusion

In this chapter, several feature types on both panchromatic and multispectral images of colon biopsies were compared. A four-class classification was computed after the feature extraction and selection process on two separate datasets. The experimental results demonstrate a clear improvement of the algorithms performance for every texture features and the classifiers used when a multispectral image is used instead of a panchromatic one. This result confirms the higher discriminative power of multispectral imaging over panchromatic imaging.

The second main conclusion that can be drawn from this study is the superiority of the LBP features over both Haralick and the LIOP counterparts for colorectal and prostate tumour discrimination.

Both of these results show that the best classification performance is achieved with the large feature vector and feature selection rather than smaller data.

In the next chapter, a better way to exploit the multispectral texture information will be investigated.

Chapter 5

Multispectral LBP Texture Feature

5.1 Introduction

In the previous chapter, it was concluded that multispectral data allowed for a more accurate classification of the prostate and colorectal tumour tissues. However, the previously used system does not consider the inter-band spectral information. We suspect that more information can be extracted from the multispectral image by taking into account this inter-band information, in order to further improve the classification accuracy.

In this chapter, based on the published papers [3, 1], a novel multispectral LBP texture feature approach is investigated. First, a brief review of the existing LBP features is carried out. Then, the proposed multispectral LBP is presented. Two different classification methods are considered and assessed. The first system uses a bag-of-features (BoF) scheme while the second imple-

ments a stacked generalisation framework. These two classification methods are detailed in the chapter. Finally, the experiments carried out are explained and their results are analysed.

5.2 Feature Extraction using LBP Approach: A Review

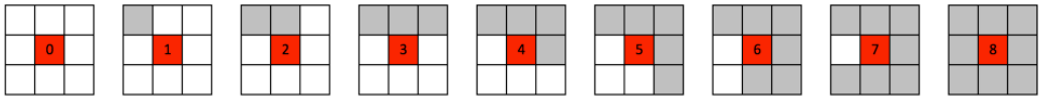
The conventional LBP was presented in Section 4.2.2. In this section its rotation invariant and 3D variants are discussed.

5.2.1 Rotation Invariant Uniform LBP

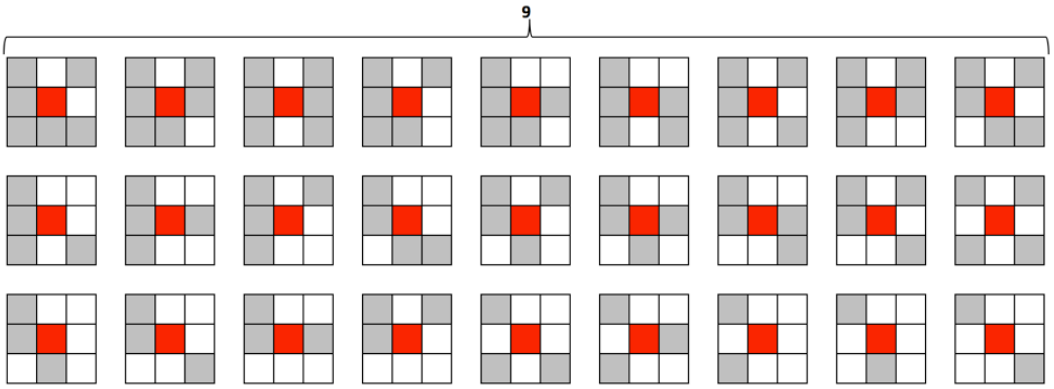
A rotation invariant LBP, referred to as LBP^{riu2}, using uniform patterns has also been proposed as illustrated in Figure 5.1a. They operate as templates for microstructures such as bright spot (0), flat area or dark spot (8) and edges of varying positive or negative curvature (1-7) [102]. These structures define a uniformity measure U corresponding to the number of transitions in the pattern as follows:

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} |s(g_p - g_c) - s(g_{p-1} - g_c)|. \quad (5.1)$$

Figure 5.1a shows the 9 patterns with a U measure of at most 2 when the 27 other patterns shown of Figure 5.1b have a uniformity measure of at least 4. Therefore, patterns having $U(LBP_{P,R}) \leq 2$ are said to be uniform. The following operator defines a grey-scale and rotation invariant texture descrip-



(a) Uniform LBP patterns and their corresponding labels



(b) Non-uniform LBP patterns

Figure 5.1: The 36 unique possibilities for a circular symetric set of LBP patterns and their corresponding labels for rotation invariant, uniform LBP. The red squares correspond to the central pixel, the white and grey squares represent the 0 and 1 bits in the 8-bits output of the operator. The numbers are the unique $LBP_{P,R}(x)$ labels.

tion [102]:

$$LBP^{riu2}_{P,R}(x) = \begin{cases} \sum_{p=1}^{P} s(g_p - g_c) \; if \; x \leq 2, \\ P+1 \qquad otherwise. \end{cases} \tag{5.2}$$

In this way, $P+1$ uniform patterns are assigned to a unique label corresponding to the number of 1 bits in the pattern while the non-uniform patterns are grouped under the same category. The final texture feature used is a histogram of $P+2$ bins generating all the $LBP^{riu2}_{P,R}$ outputs accumulated over the image.

This form of LBP seems more adapted to the problem at hand because of the rotation invariance it provides. Indeed, in the case of histopathology, sample orientation and cells direction are not relevant criteria to consider for classification because they vary independently to the sample's class. A second

advantage of this $LBP^{riu2}_{P,R}$ over a conventional $LBP_{P,R}$ is its smaller size thus making it faster to process in a classification phase.

5.2.2 3D-LBP

Since multispectral images are 3D data the conventional LBP concept needs to be modified to deal with this datatype. In the literature, two methods are usually described when dealing with 3D images for applications such as video processing and face recognition [123]. The proposed method is inspired from Volume Local Binary Pattern (VLBP) and LBP-TOP (for Local Binary Pattern-Three Orthogonal Plan) [123]. VLBP and LBP-TOP are briefly discussed in this section. To extend LBP to dynamic texture analysis, Zhao *et al.* define a neighbourhood as the joint distribution of $3P + 3$ image pixels where P is the number of neighbours on one frame as shown on [123]. A similar technique to the conventional LBP can be applied and a VLBP is defined as follows [123]:

$$VLBP_{P,R}(x) = \sum_{p=1}^{3P+2} s(g_0 - g_p)2^{p-1}. \tag{5.3}$$

The VLBP local features are pooled into a histogram of size 2^{3P+2}. This histogram's size increases very rapidly when the number of neighbours, P, grows and may become very computationally expensive. On the other hand, using a small P may lead to a loss of some critical information for diagnosis purpose. To address this issue, a LBP-TOP feature is proposed by considering three orthogonal planes intersecting on a central pixel as described in [123]. The technique computes a two-dimensional LBP on each of these plans and concatenates the output histograms which will be of size $3 * 2^P$ instead of 2^{3P+2} previously used. In [123], the circles are considered in the time dimension

because this LBP-TOP is meant to be applied on video processing so the motion direction of texture is unknown.

5.3 The Proposed Multispectral Multiscale LBP Texture Feature

In the proposed technique the third dimension is spectral (not temporal), therefore no texture motion is considered. Consequently, unlike in the aforementioned 3D-LBP variants, a neighbourhood of only P points in the spatial plan and P_λ on a straight line in the spectral dimension intersecting the spatial plan at the central pixel was considered as shown in Figure 5.2 where $P_\lambda = 2$. As explained above, this technique is adopted to make the LBP rotation invariant in the spatial dimensions while still using the same U measure described in 5.2 in the XY plan.

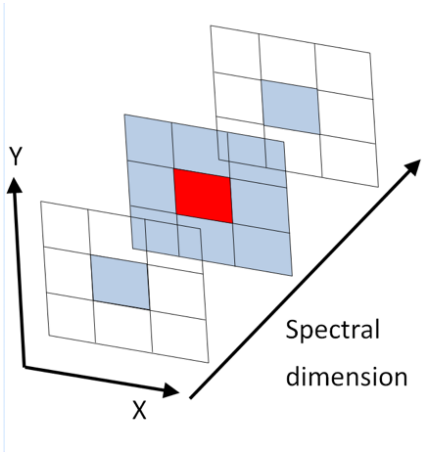


Figure 5.2: Multispectral LBP descriptor: the neighbourhood considered for multispectral LBP. X and Y being the spatial dimensions. Each tile represents a pixel. The red tile is the central pixel considered, and the blue tiles are the pixel considered in the neighbourhood.

The key idea is to assign the $LBP^{riu2}_{P,R}$ patterns to different categories depending on the P_λ pixels in the neighbouring plans. On top of the $LBP^{riu2}_{P,R}$ computed using equation 5.2, the $LBP^{\lambda}_{P_\lambda,R}$ is calculated using the following equation:

$$LBP^{\lambda}_{P_\lambda,R}(x) = \sum_{q=1}^{P_\lambda - 1} s(g'_q - g_c)2^q, \tag{5.4}$$

where, g'_q is the pixel value in the pixel of plan q aligned to the central pixel. The $MMLBP_{P,P_\lambda,R}$ is defined as follows:

$$MMLBP_{P,P_\lambda,R} = LBP^{riu2}_{P,R} + (P+1)LBP^{\lambda}_{P_\lambda,R}. \tag{5.5}$$

The $MMLBP_{P,P_\lambda,R}$ outputs are then pooled into a histogram of size $(P + 2) * 2^{P_\lambda}$. It is worth noting that the scale is controlled by $R \in [1..N_{scale}]$. As a result, the histograms built from each scale are concatenated to form the MMLBP. Each scale is built separately as described in Figure 5.3. The resulting vectors can either be concatenated or fed to different classifiers depending on the classification scheme chosen. This is detailed in Sections 5.4 and 5.5.

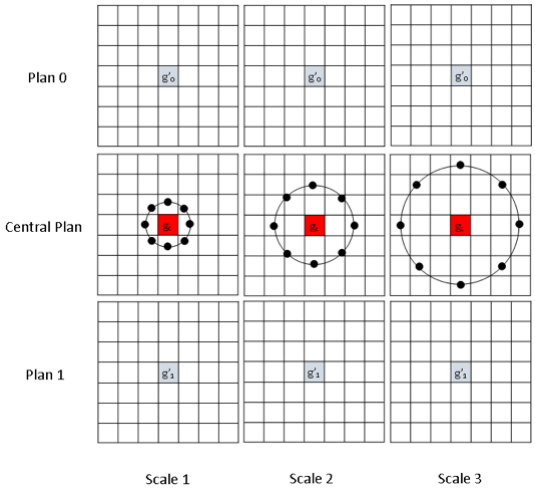


Figure 5.3: Multiscale neighbourhood for MMLBP

5.4 MMLBP System with BoF Classification Scheme

For the last fifteen years, BoF approaches have proved to be very efficient in various computer vision applications [124, 125, 126, 127]. It is based on a methodology used in text classification where the Bag-of-Words (BoW) approach and textons are commonly used. In text documents classification, a dictionary of words is built and the words frequency from this dictionary is quantified for each text document in order to classify them. The BoF approach was thought of as an analogy to the aforementioned technique. In this representation, a dictionary of image features is built in order to recognise image feature patterns. Unlike image segmentation, objects in an image are not identified. Small regions of the image are instead characterised in order to represent its content. Therefore, the technique shows very good adaptiveness to the dataset used by identifying the particular features relevant to the complete dataset. The reason being that each pattern used for describing an image comes from the analysis of the whole dataset. The analysis in small image regions also makes this approach robust to translations and rotations as well as occlusion, making it ideal for medical imaging applications [124, 128, 129].

This section presents a system based on MMLBP texture extraction and using histograms of codebooks as image descriptors for classification. First, the image descriptors extraction is discussed. Then, the classification method is addressed.

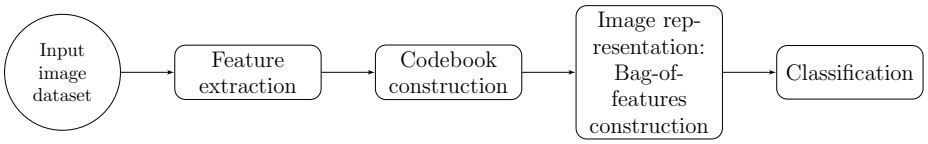


Figure 5.4: BoF representation steps.

5.4.1 Image Descriptor: Histograms of Codebooks

The BoF framework is based on an analogy with the BoW framework. A dictionary - or codebook - of visual words - called codewords - that represent the most characteristic patterns of dataset is built. In order to construct the image descriptor, the occurrence of each codeword is computed and a histogram of their occurrence is created.

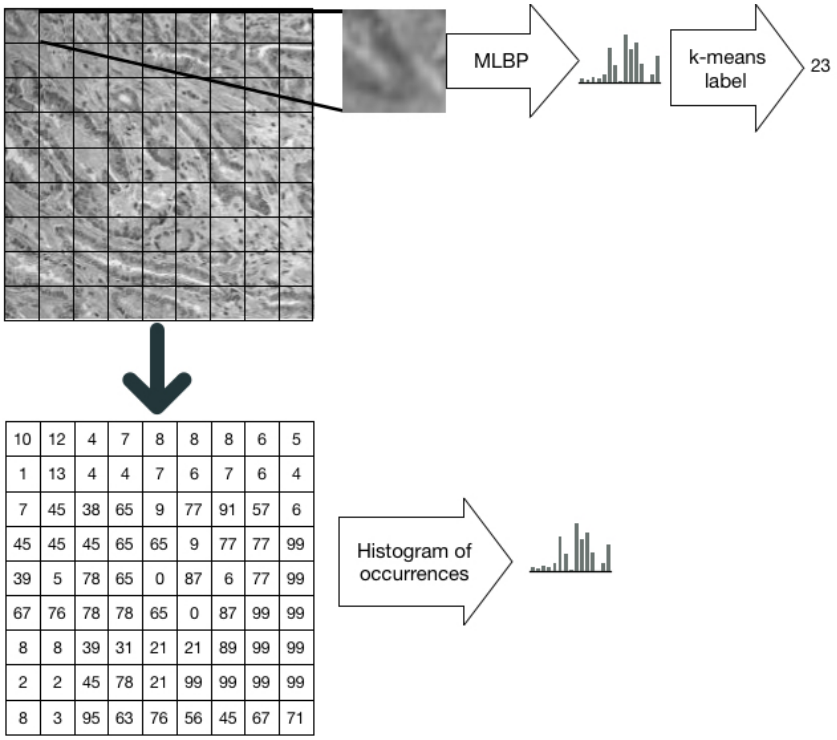
Figure 5.4 shows the four steps of image classification using a BoW framework described by Csurka *et al.* in [126].

5.4.2 BoF Framework

The image descriptor extraction is described in Figure 5.5.

A block-based image processing is conducted in this system. Images are divided into small overlapping blocks and the feature vectors are extracted from each block. In this case, the MMLBP texture feature described in Section 5.3 is used. Each block is therefore represented by a 40-bin histogram characterising its multispectral texture.

After this block-wise feature extraction, a k-means clustering of all the block descriptors from all the images is carried out. This operation clusters the collection of image blocks into k groups.



10	12	4	7	8	8	8	6	5
1	13	4	4	7	6	7	6	4
7	45	38	65	9	77	91	57	6
45	45	45	65	65	9	77	77	99
39	5	78	65	0	87	6	77	99
67	76	78	78	65	0	87	99	99
8	8	39	31	21	21	89	99	99
2	2	45	78	21	99	99	99	99
8	3	95	63	76	56	45	67	71

Figure 5.5: Image descriptor extraction framework.

Each cluster member is then identified with the cluster's centroid and each centroid is assigned a label. This operation results in the generation of the codebook that will be used for the dataset image description. For each image, the corresponding label is assigned to each block and the frequency of each label is then calculated. The final image descriptor is the normalised histogram of label occurrences.

Multiscale BoF System

For a multiscale system, described in Figure 5.6, the MMLBP features are extracted for different R parameters of the multispectral LBP texture feature as described in Section 5.3. A different codebook is then created for each scale R with their respective multispectral LBP features using the aforementioned technique - Section 5.4.2. This collection of codebooks is named a multiscale

codebook. Finally, the histograms of occurrences are computed for each scale and concatenated into one sole final image descriptor as shown in Figure 5.6.

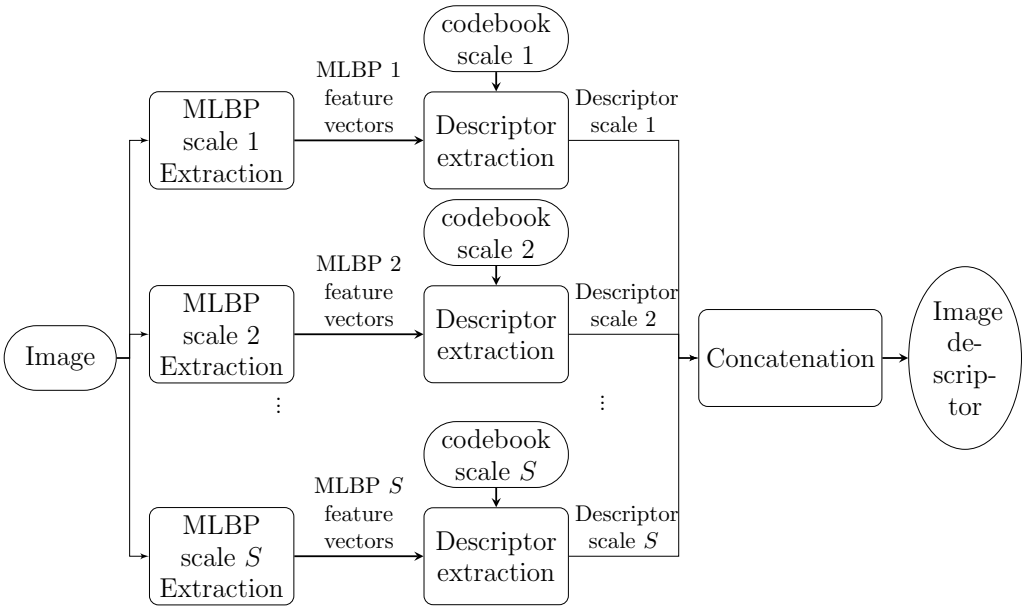


Figure 5.6: Block diagram of the multiscale MLBP feature extraction.

Bagged Codebooks

In the technique described in Section 5.4.2, the whole collection of blocks from all the images of the dataset is used at the clustering phase for codebook creation. This very large number of feature vectors generated in this manner would require a very large memory space and may also lead to overfitting. To address this issue, a codebook is created from a number N of randomly selected features from all the images as shown in Figure 5.7. More precisely, N blocks are randomly selected from the collection of image blocks in the dataset and MMLBP features are extracted from these particular blocks. Clustering is then carried out using a k-means algorithm on this subset of selected feature vectors, resulting in the creation of a codebook. For the remaining blocks, the MMLBP feature vectors are extracted and assigned to the cluster with their

closest centroid. As a result each block is represented by a label and the label occurrences can be computed for image descriptor construction purposes.

The random selection of features vectors for codebook generation poses a challenge of class representation. A bagging ensemble method is chosen to overcome this issue. This technique, introduced by Breimann [130], is based on the idea that using multiple versions of a predictor and aggregating their results increases the final prediction accuracy. In order to have an accurate representation of the feature space a number M of codebooks have to be created as shown in Figure 5.7. As a matter of fact, the theory of bagging is based on the fact that the predictor models trained on different data will not always lead to the same results due to the variation of learning sets [131]. By training M prediction models on different learning sets and aggregating their results, this variation is therefore compensated. This number of codebooks M will be optimised as a system hyperparameter during experimentation.

5.4.3 Classification

For the training phase described in Figure 5.8, multispectral LBP are extracted for each scale for every block of the training set images. The M multiscale codebooks previously created as mentioned in Section 5.4.2 are used for multiscale descriptor extraction. Then, a classification model is trained for each multiscale codebook. In this system, the classifier used is the SVM.

In the testing phase of the system, which is shown in Figure 5.9, the feature vectors are extracted from each block of the image and used to create a descriptor of the image by creating a histogram based on each multiscale codebook with the same approach used at the training stage. In this way, M descriptors

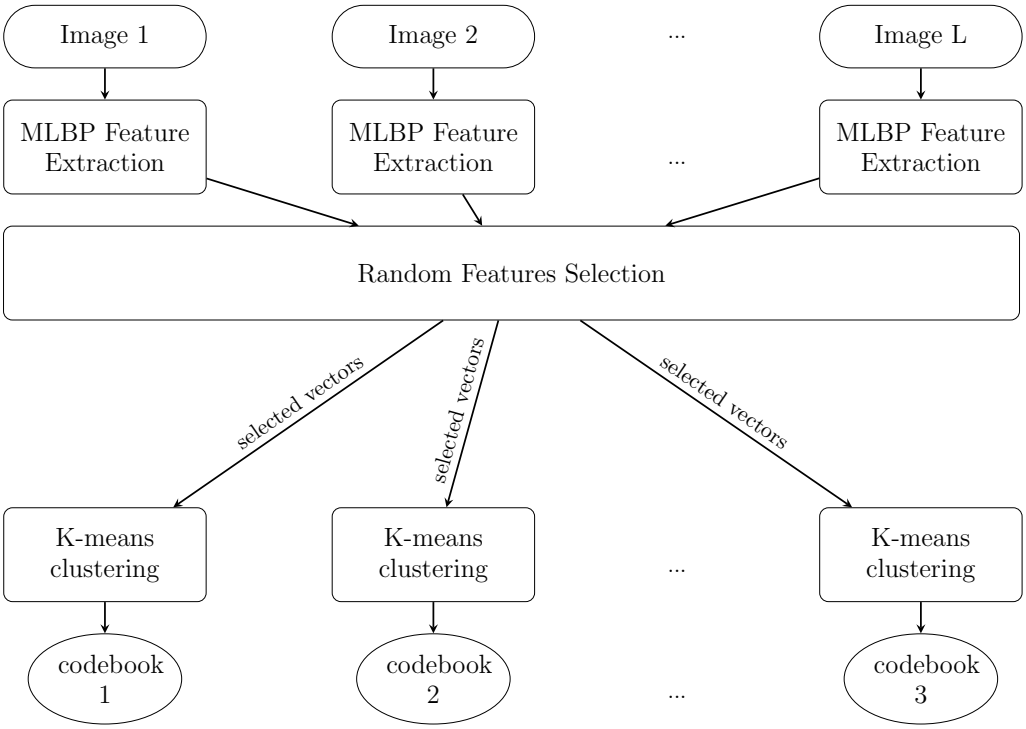


Figure 5.7: Block diagram of the bagged codebooks generation.

are collected for the input image. They are fed to their respective SVM predictor model and the class receiving the most votes from the M predictors is the system's predicted class for the input image.

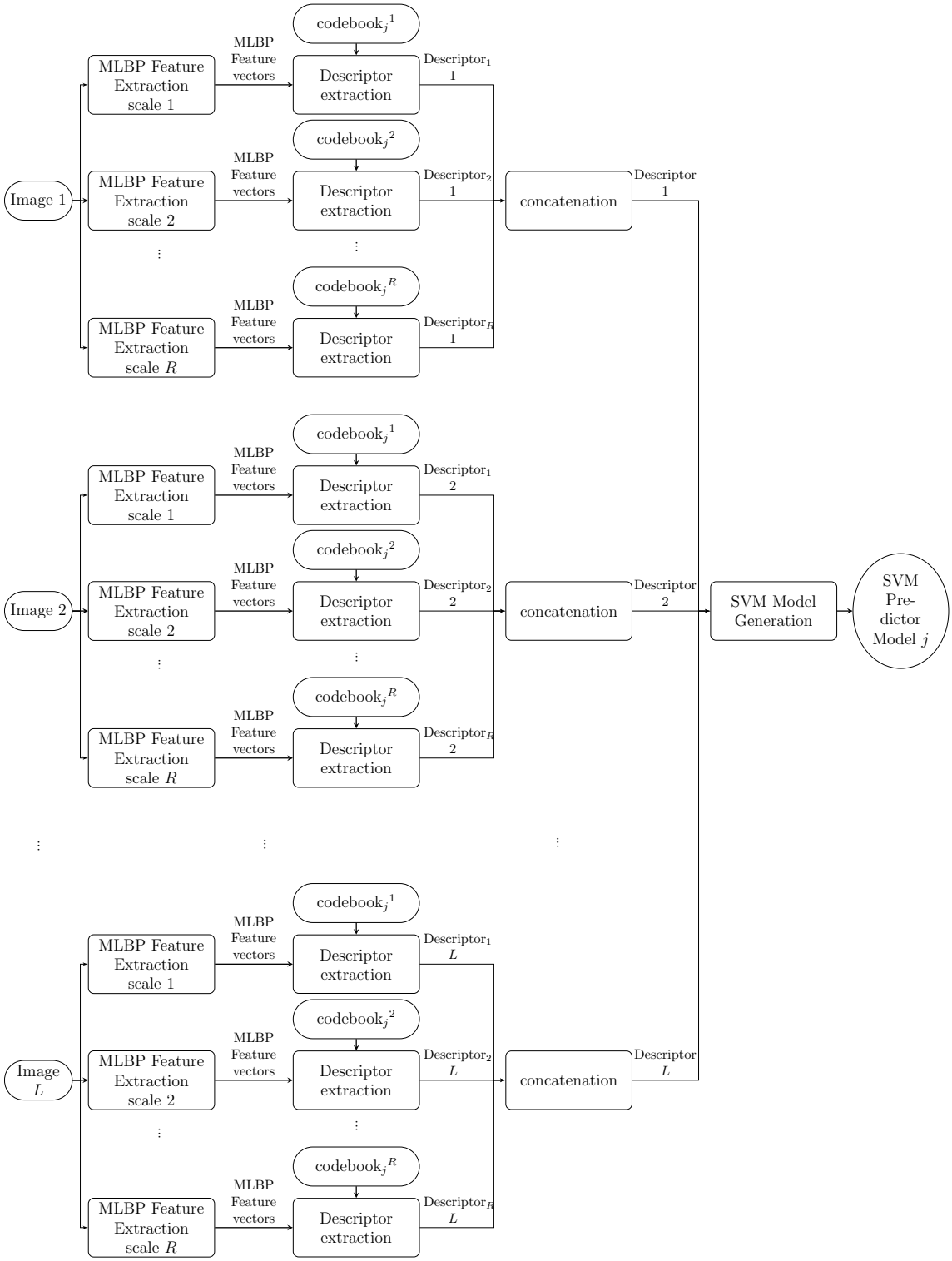


Figure 5.8: Block diagram of the proposed system's training phase.

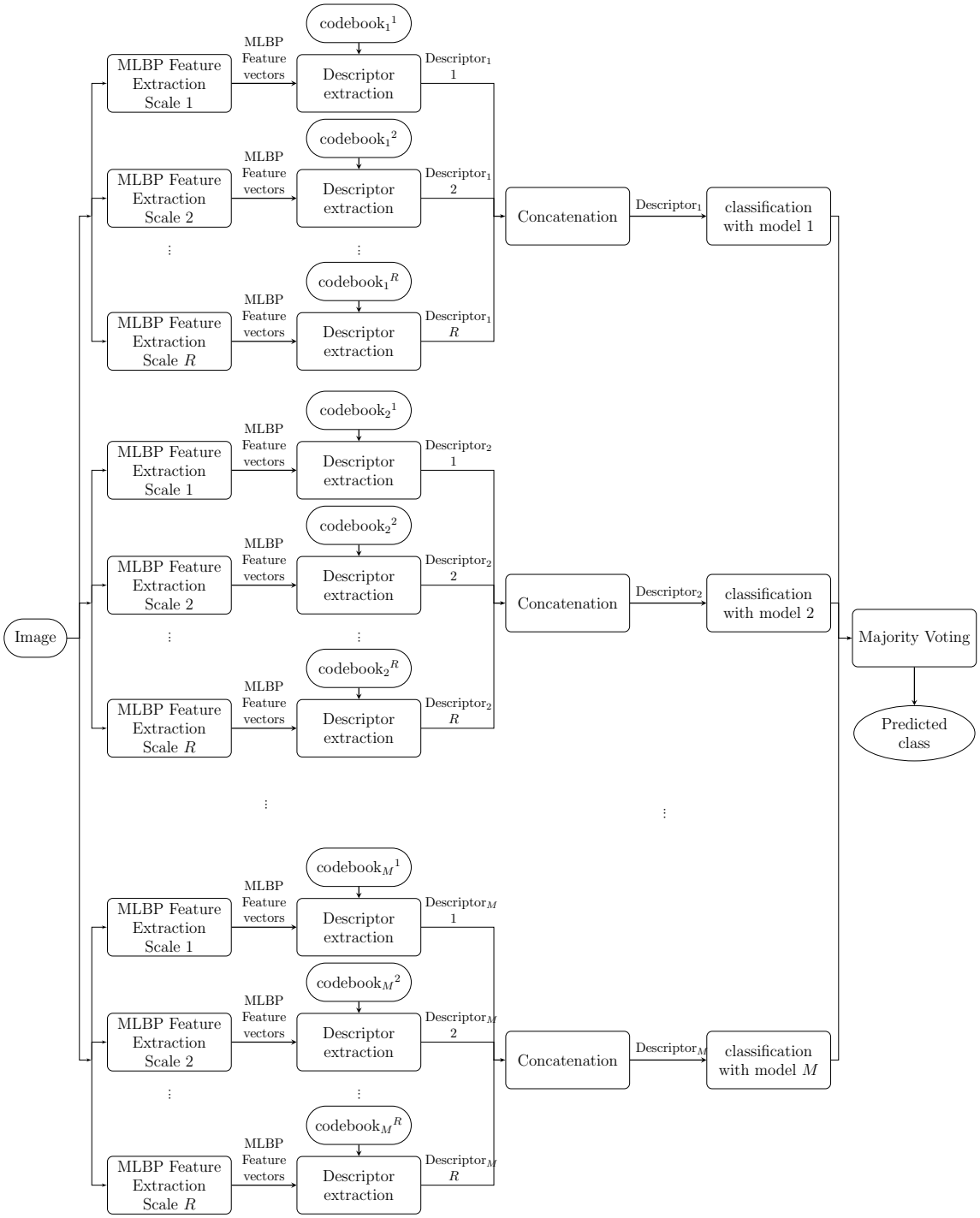


Figure 5.9: Block diagram of the proposed system's testing phase.

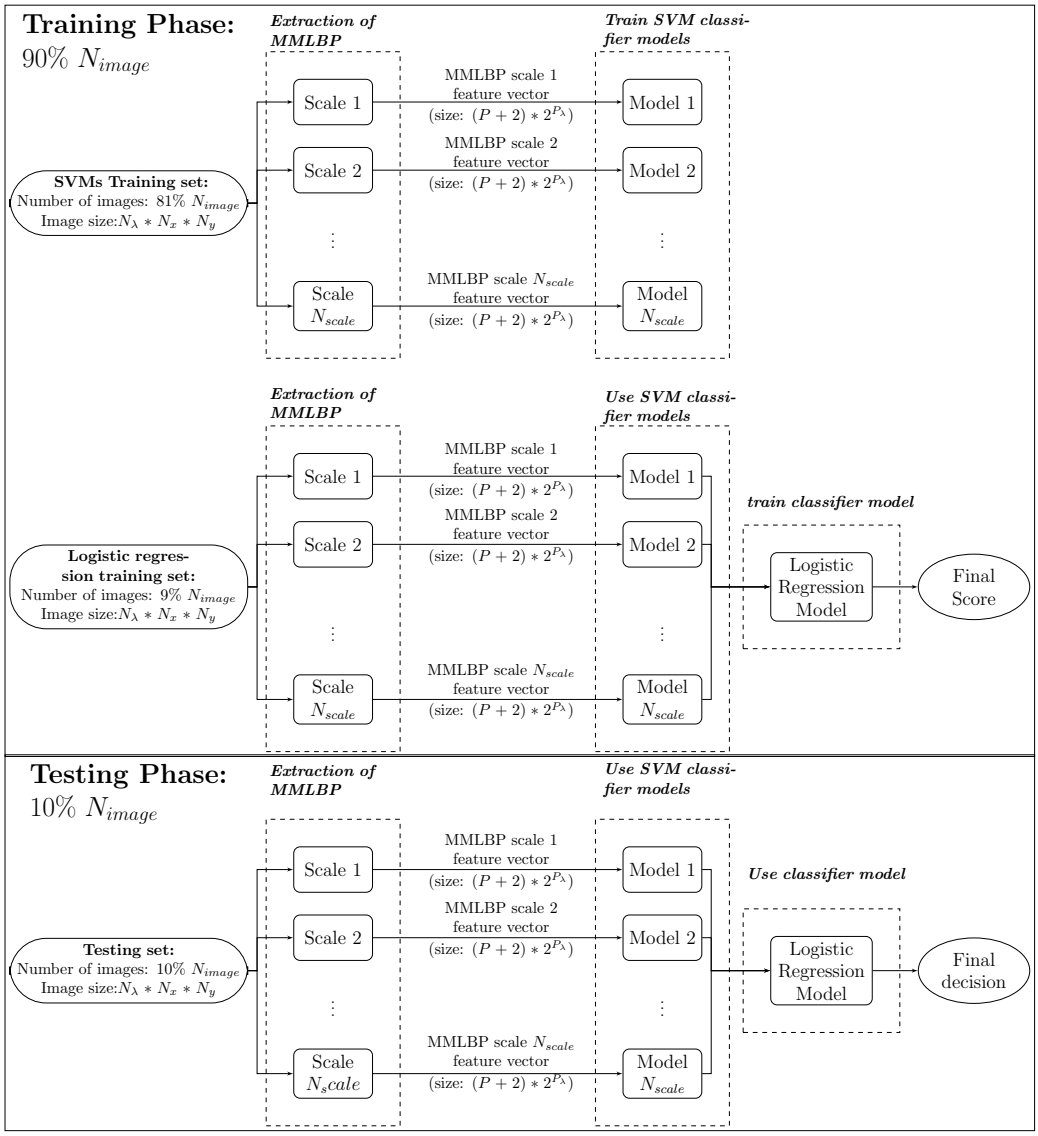


Figure 5.10: Block diagram of stacking training and testing with MMLBP texture features. N_{image} represents the number of images in the dataset; N_x, N_y are the number of lines and columns in each image, respectively, and N_λ is the number of spectral bands.

5.5 MMLBP System with Stacked Generalisation Classification Scheme

As illustrated in Figure 5.10, the proposed system is composed of two main stages. First, MMLBP features are extracted and, then, an Independent Com-

ponent Analysis (ICA) is performed to reduce the dimensionality of the feature space. In the second phase, a stacked generalisation employing the Support Vector Machine classifier is used at the matching stage.

5.5.1 Dimensionality Reduction using ICA and Classification using SVM

ICA

In order to address the curse of dimensionality problem and hence reduce the learning cost, the ICA is applied before classification. In contrast to the more widely used PCA, this technique presents the advantage of being able to decorrelate the signal and reduce statistical dependencies between the features as much as possible [132]. In fact, it could be seen as a version of PCA that defines orthogonal directions. The ICA transformed data are computed using only the training data of the SVM classifier. The testing data are projected to the new basis before classification. The number of components used for classification is optimised as described in Section 5.6.4.

In the ICA, the main goal is to represent the data with minimizing the statistical dependence of the components [133, 134, 135].

Definition 1. ***Statistical Independence:*** *Let $y_1, y_2, \ldots, y_m$ a set of m zero-mean random variables with a joint density $f(y_1, y_2, \ldots, y_m)$. The variables y_i, $i \in [1 \ldots n]$, are mutually statistically independent if:*

$$f(y_1, y_2, \ldots, y_m) = f_1(y_1) f_2(y_2) \ldots f_m(y_m), \tag{5.6}$$

where, $f_i(y_i)$ is the marginal density of y_i.

Statistical independence is therefore a very strong condition requiring an infinite amount of data. Consequently, in practice, a proxy measure is used, usually under the form of a function to maximise.

Definition 2. ***ICA:*** *Let* $\mathbf{x}$ *be a* m*-dimensional random variable. The ICA of* $\mathbf{x}$ *consists of finding a* $m \times n$ *matrix* $\mathbf{W}$ *so that:*

$$\mathbf{s} = \mathbf{W}^T\mathbf{x}, \tag{5.7}$$

where, $\mathbf{s} = (s_1, s_2, \ldots, s_n)^T$ *and its components* s_i *are considered mutually statistically independent.*

A widely used algorithm for computing the ICA is the FastICA introduced by Hyvarinen [134]. It is a fixed-point algorithm and works on prewhitened data. Prewhitening the data consists of transforming the data to give them the same characteristics as white noise. For this purpose, a linear transformation resulting in uncorrelated data with a variance equals to one is applied to the centred data [134]. The FastICA is based on a non-Gaussianity measure as a proxy for statistical independence. This non-Gaussianity is measured with a non-quadratic non-linear contrast function $f(u)$ and its first and second derivatives, $g(u)$ and $g'(u)$. In [134], Hyvarinen shows that $f(u) = -e^{-u^2/2}$, $g(u) = ue^{-u^2/2}$ and $g'(u) = (1 - u^2)e^{-u^2/2}$ are adapted for problems where robustness is very important. The steps of the FastICA algorithm are described in algorithm 1.

After experimenting with the PCA for dimentionality reduction, the results obtained with ICA in agreement with the theory and show an improved accuracy due to the statistical independence of the components selected by the algorithm. As a consequence, The ICA is selected as a dimensionality re-

Algorithm 1: FastICA

FastICA $(\mathbf{X}, n)$;
Input : non-negative integer n: number of desired components
Input : whitened matrix $\mathbf{X} \in \mathbb{R}^{m \times l}$, where each one of the l column represents a m-dimensional sample ($n \leq m$)
Output: $\mathbf{W} \in \mathbb{R}^{m \times n}$ a matrix where each column projects $\mathbf{X}$ into the independent components space
Output: $\mathbf{S} \in \mathbb{R}^{n \times l}$: the independent components matrix, with l n-dimensional columns, each representing a sample
for $p \leftarrow 1$ **to** n **do**
  $\mathbf{w_p} \leftarrow$ Random vector of length m;
  while $\mathbf{w_p}$ *not converging* **do**
    $\mathbf{w_p} \leftarrow \frac{1}{n}\mathbf{X}g(\mathbf{w_p}^T\mathbf{X})^T - \frac{1}{n}g'(\mathbf{w_p}^T\mathbf{X})\mathbf{1}\mathbf{w_p}$; `// 1 is a` n`-dimensional vector of ones`
    $\mathbf{w_p} \leftarrow \mathbf{w_p} - \sum_{j=1}^{p-1} \mathbf{w_p}^T\mathbf{w_j}\mathbf{w_j}$; `// Independence to the other components`
    $\mathbf{w_p} \leftarrow \frac{\mathbf{w_p}}{||\mathbf{w_p}||}$; `// Vector normalisation`
  end
end
return $\mathbf{W} = [\mathbf{w_1}, \mathbf{w_2}, \ldots, \mathbf{w_n}]$;
return $\mathbf{S} = \mathbf{W}^T\mathbf{X}$;

duction technique. The classifier used in this system is based on the SVM classifier described in Chapter 4. The kernel parameters are optimised using a grid-search method which is detailed in Section 5.6.4. In order to find the appropriate compromise between the sizes of training and testing datasets and hence avoid over-fitting that might be caused by a leave-one-out technique, a 10-fold cross-validation is used as mentioned in Section 5.6.3. The OvsA scheme is used to build the multiclass classifier.

5.5.2 LR for Stacked Generalisation

Stacked generalisation (or stacking) is an ensemble method for classification [136]. It uses the output of a first layer of classifiers as inputs to another classifier - called meta-classifier - for the final decision. In this chapter, this system is used to fuse the different scales of multispectral LBP texture feature at score level.

Figure 5.10 shows the two steps of training and testing for the stacking algorithm. A LR model is used as a meta-classifier for its relatively low computing cost. The first layer of classifiers is composed by SVM classifiers with a Gaussian kernel as described in Section 5.5.1. In addition to a 10-fold cross-validation carried out at the meta-classifier level, an internal cross-validation of the training data is implemented in order to prevent bias and to improve stability of the different classifiers.

5.6 Experiment and Setup

5.6.1 Experiments

In order to assess the performance of the proposed MMLBP texture feature, different classification frameworks are tested.

The proposed systems are compared with the results given by the algorithm described in Chapter 4 by using a conventional LBP extracted from a panchromatic image that is generated by averaging the spectral bands of the multispectral image. Similarly, it is compared to another variant of LBP adapted to multispectal images, which consists of extracting LBP histograms from each band and then concatenating them to generate a final descriptor. This method, is referred to as the **concatenated LBP**. A system using the concatenated MMLBP features coupled with a SVM classifier is also tested. It is denoted **concatenated MMLBP**. It is worth mentioning that these LBP variants are used with an SVM classifier for a fair comparison. For the same reason, they are also applied using the same number of scales N_{scale}. Many authors use GLCM texture features, described in Chapter 4.

The proposed system using the BoF framework described in Section 5.4.2 is referred to as **BoF**. **BoF multiscale** refers to the multiscale version of this system.

The proposed system using stacked classification of GLCM features combined with MMLBP features as shown in Figure 5.11 is denoted as **stacked MMLBP + GLCM**. Its results are also compared to the ones given by MMLBP alone - denoted as **stacked MMLBP**.

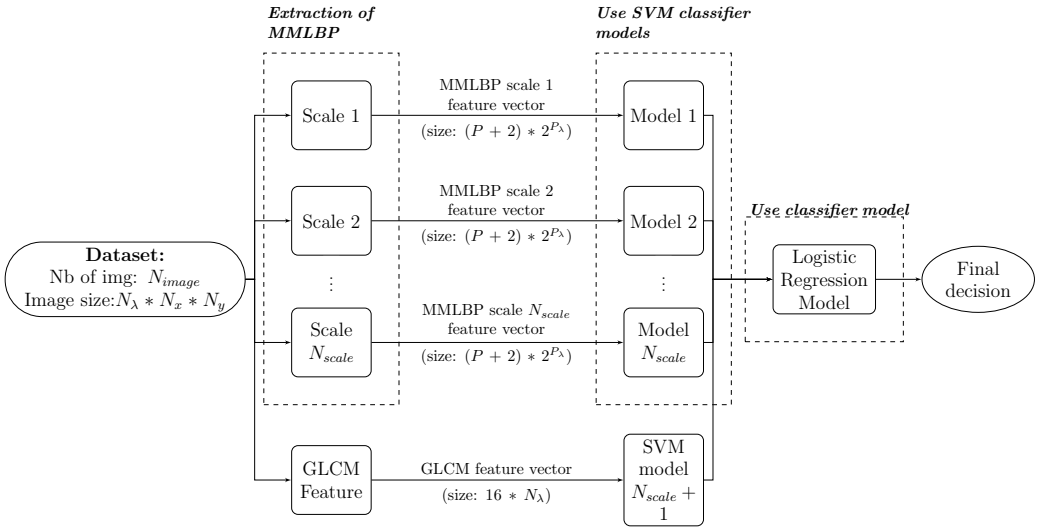


Figure 5.11: Block Diagram of the proposed stacked MMLBP + GLCM.

In the second set of experiments, the impact of spatial resolution variations of the performances is discussed.

The algorithms are also compared against different algorithms from the literature that used the prostate dataset or that can be implemented and tested on the prostate dataset. An adapted version of Masood *et al.*'s algorithm [45] to the multiclass problem is implemented. In this method the authors use the GLCM features after segmentation of the image to train an SVM classifier. The methods presented are also compared against Khelifi *et al.*'s results [39]. The authors define a multispectral form of the GLCM before extracting the GLCM features. Finally the results shown in [19] are used for comparison purposes. In [19], Tahir *et al.* describe a Round-Robin Tabu search algorithm for prostatic tumor classification.

Finally, the colorectal dataset is used to assess the usefulness of the IR spectrum. The stacked MMLBP + GLCM algorithm is applied to the visible (Vis) and IR bands of the dataset separately and then to the images with both parts of the light spectrum combined. The results are then compared.

5.6.2 Evaluation Measures

In order to avoid accuracy variations, the cross-validation is run ten times and the accuracy is averaged. The standard deviation is calculated on the mean accuracies of each cross-validation.

In addition to the accuracy and the standard deviation, the ROC curve and the AUC and the confusion matrix are also computed and used to assess the performances of the proposed algorithm. These performance measures are useful metrics to allow for a better understanding of what each class captures before OvsA combination to obtain the overall accuracy.

5.6.3 Training Procedures

BoF

A 10-fold cross-validation scheme is adopted in order to improve the generalisation estimation and reduce the standard deviation on this estimation. 10 folds are chosen as the best compromise between the sizes of the training and test datasets. The test dataset needs to be large enough so that the variance on its estimation is as small as possible. However, the training set also needs to have a sufficient number of examples for the model to not be in the underfitting regime.

Stacking MMLBP + GLCM

As illustrated by Figure 5.10, the double 10-fold cross-validation run on the datasets means that, for each experiment, 90 % of the dataset is used for

Table 5.1: Number of images used in each phase for each the tested dataset.

Data-set	Prostate	Colorectal
SVMs training set	415	518
Logistic regression training set	46	58
testing set	51	64
dataset size	512	640

training the LR classifier model and the remaining 10 % are used for the testing phase. 90 % of this training set (or 81 % of the total dataset) is used for training the SVM models and in the remaining 10 % of the training set (or 9 % of the whole dataset), the trained SVM models are used to train the LR model. Table 5.1 displays the SVMs and LR training sets and the testing set sizes for each dataset.

5.6.4 Parameters Tuning

BoF

A number of hyperparameters need to be tuned for this algorithm:

- The size of codebooks which corresponds to the number of clusters k in the k-means clustering. It also determines the size of the image descriptor (descriptor size = $k*$number of scales),
- The number of image features selected for codebooks generation N,
- The number of different codebooks generated at each scale M,
- The SVMs kernel parameters C and γ.

A grid-search method is adopted to find the optimal combination of these hyperparameters:

$$k = 20 * i, \text{ with } i = [|4 : 12|],$$

$$N = 500 * i, \text{ with } i = [|1 : 10|],$$

$$M = 5 * i, \text{ with } i = [|1 : 10|],$$

$$C = 10^i, \text{ with } i = [|-3 : 3|],$$

$$\gamma = 10^i, \text{ with } i = [|-3 : 3|].$$

For each combination of the parameters in these intervals, the accuracy is calculated and averaged with a 10-fold cross-validation. The parameters giving the maximum average accuracy are then chosen as the model parameters.

Stacking MMLBP + GLCM

As discussed previously, a total of 3 parameters need to be optimised for each SVM classification: the number of components selected in the ICA, and the C and γ parameters of the SVM kernel. A 3D grid-search was performed with the following parameters, with a step equals to 1:

$$C = 10^i, \text{ with } i = [|-3 : 3|],$$

$$\gamma = 10^i, \text{ with } i = [|-3 : 3|],$$

$$N_{comp} = 10 * i, \text{ with } i = [|1 : 50|].$$

For each combination of the parameters in these intervals, the accuracy is calculated and averaged with a 10-fold cross-validation. The parameters giving the maximum average accuracy are then chosen as the model parameters.

5.7 Results and Discussion

5.7.1 Proposed Algorithm Discussion

Table 5.2 shows a comparison of the classification accuracies obtained using different features and classification methods. First, a conventional LBP followed by a SVM classification is performed and an accuracy of 88.9 % is found for the prostate dataset while this algorithm was 51.3 % accurate for the colorectal dataset. This shows that this option is not robust to the data. When using a concatenated version of multispectral LBP followed by an SVM classification, the results are improved and accuracies of 92.4 % and 88.2 % are achieved on the prostate dataset and the colorectal dataset, respectively. This shows how the multispectral information improves the classification accuracy. However, there still is a high variation of accuracy between both datasets which highlights a lack of robustness to the data. The added discriminative power of the inter-band information is proven by the increased accuracy when using concatenated MMLBP. It is indeed improved by 1.8 pp and 0.9 percentage points (pp) for the prostate and colorectal datasets, respectively.

With BoF framework using a single scale texture features, the estimated accuracy is higher than concatenated MMLBP (which does have the multiscale information). However, the standard variance computed shows that the differences between these estimated accuracies are within the margin of error.

Therefore, it can be considered that there is no improvement from using the single scale BoF as opposed to the concatenated MMLBP. Nonetheless, the BoF multiscale is a fairer comparison to the concatenated MMLBP as the features in this system also capture the multiscale information and only the classification method changes. An increase accuracy of 96.5 % and 91.2 % is observed when using the multiscale information on the prostate and colorectal datasets, respectively. It can be noticed that the standard variations observed with both BoF and BoF multiscale are higher than the ones observed with the previous simple classifiers.

When using stacked MMLPB, the results are further improved and an accuracy of 99.2 % and 98.9 % on the prostate and colorectal datasets, respectively, therefore demonstrating the robustness of the proposed algorithm. This can be explained because the stacking method selects the best features for classification and discards the features that drop the accuracy, which is independent to the data. When GLCM texture features are combined to the MMLBP texture features the results are improved by 0.3 pp and 0.6 pp for the prostate and colorectal datasets, respectively.

Finally, the multispectral spectral information adds significant improvement over the conventional LBP as illustrated by the performance of the concatenated LBP method. However, the multispectral information is better captured by the MMLBP texture feature as demonstrated by the improvement observed with the concatenated MMLBP. Furthermore, the stacking classification process enhances the performance further as demonstrated by the results of the stacked MMLBP compared to the concatenated MMLBP and the BoF multiscale. The reason for a lower performance of the BoF algorithm probably comes from the difficulty to find optimal parameters for the SVMs. Each of

Table 5.2: Accuracy (in %) comparison of different feature extraction and classification methods

Dataset	Prostate	Colorectal
Conventional LBP	88.9 ± 0.6	51.3 ± 0.7
Concatenated LBP	92.4 ± 0.4	88.2 ± 0.5
Concatenated MMLBP	94.2 ± 0.3	89.1 ± 0.4
BoF	95.0 ± 0.7	88.7 ± 0.8
BoF multiscale	96.5 ± 0.8	91.2 ± 0.9
Stacked MMLBP	99.2 ± 0.3	98.9 ± 0.4
Stacked MMLBP + GLCM	99.5 ± 0.3	99.5 ± 0.1

them being trained on a different random subset of the training set data, their optimal parameters vary. Being able to have finely tuned SVMs parameters could mean an increased performance of the BoF system.

For the Stacked MMLBP + GLCM, Figure 5.12 and 5.13 displays the ROC curves and shows the AUC for the different classes in a binary classification following the OvsA scheme. Tables 5.3 and 5.4 show the confusion matrices obtained with BoF multiscale. This is done to assess the positive and negative false alarm rates for each class. In both algorithms, it can be observed that the classification for the cancerous class is always the one performing best. This can be explained by the specific characteristics and features present in the images of this class. It corresponds to the binary classification cancerous vs non-cancerous tissues.

Table 5.3: Confusion Matrix of BoF multiscale for prostate dataset.

	Class BH	Class Ca	Class IN	Class Str
Class BH	119.7	0	4.6	3.7
Class Ca	0	128	0	0
Class IN	3.2	0	123.4	1.4
Class Str	6.1	0	0.0	122.9

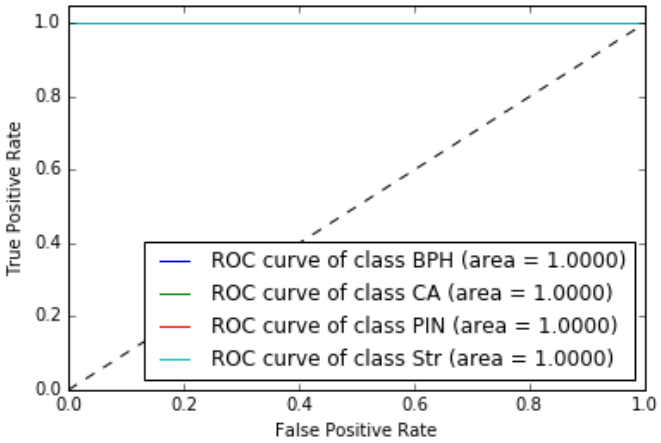


Figure 5.12: ROC for stacked MMLBP + GLCM for prostate dataset.

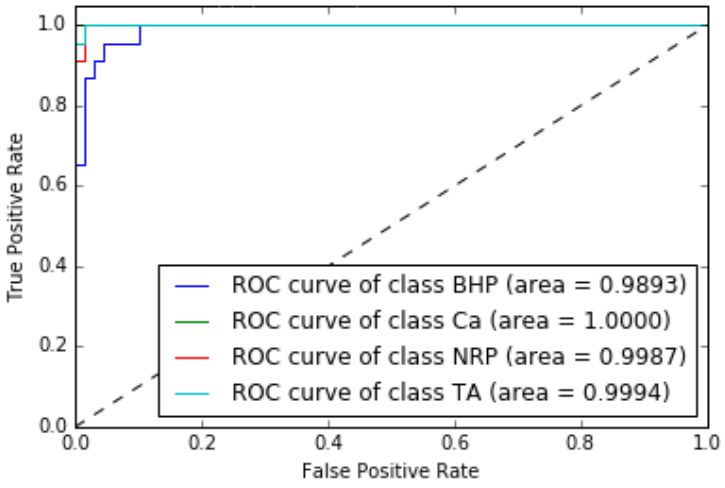


Figure 5.13: ROC for the stacked MMLBP + GLCM for colorectal dataset.

Table 5.4: Confusion Matrix of BoF multiscale for colorectal dataset.

	Class Ca	Class HP	Class TA	Class NRP
Class Ca	39.0	0	0.3	0.7
Class HP	0	38.1	0	1.9
Class TA	0.5	0.8	36.4	2.3
Class NRP	0.5	1.1	3.3	35.1

5.7.2 Impact of the Spatial Resolution

Table 5.5 shows the impact of image spatial resolution on the results of stacked MMLBP + GLCM. It demonstrates that the accuracy is marginally influenced

Table 5.5: Accuracy (in %) comparison of different spatial resolution.

Data-set	Prostate	Colorectal
Resolution 100 %	99.5 ± 0.3	99.5 ± 0.1
Resolution 75 %	99.8 ± 0.3	98.8 ± 0.4
Resolution 50 %	99.5 ± 0.3	99.4 ± 0.4
Resolution 25 %	97.6 ± 0.3	98.7 ± 0.4
Resolution 10 %	96.0 ± 0.4	96.3 ± 0.4

by the change of resolution. It varies from 99.5 % ± 0.1 for the full resolution to 98.7 % ± 0.4 for a spatial resolution of 25 % the original one for the colorectal dataset. For a resolution of 10 %, the accuracy drops to 96.3 %. The same consistency can be seen on colorectal dataset until 50 % of the original resolution then a drop by 2 points in accuracy is noticed for 25 % of the original resolution. The drop further continues with a resolution of 10 % the original one. This shows the robustness of the MMLBP algorithm presented in this paper to spatial resolution reduction until a certain percentage depending on the dataset.

5.7.3 Comparison Against Existing Algorithms

Table 5.6 depicts the performance accuracy obtained when comparing the proposed algorithms against some existing methods in the literature. For this comparison, only the prostate dataset is used as this is the only one used by other authors. Kelifi's [39] algorithm is tested on prostate dataset. Masood's *et al.* algorithm is evaluated using the prostate datasets using a multiclass classifier instead of the authors' binary classifier [45]. As can be seen in Table 5.6, the proposed method outperforms these two other algorithms in terms of accuracy. Tahir *et al.*'s algorithm is evaluated using the prostate dataset as reported by the authors who achieved a 98.9 % accuracy. The proposed

Table 5.6: Accuracy comparison to literature methods.

Method	Accuracy
Khelifi *et al.* [39] (%)	75.6
Tahir *et al.* [19] (%)	98.9
Bouatmane *et al.* [21] (%)	99.83
Masood *et al.* [45] (%)	85.1
BoF multiscale (%)	96.5 ± 0.8
Stacked MMLBP + GLCM (%)	99.5 ± 0.3

algorithms are implemented on the same dataset and the results of 99.5 % accuracy clearly show that the proposed stacked MMLBP + GLCM technique outperforms the other methods of the literature. Bouatmane *et al.* [21] claim an accuracy of 99.83 % on the same dataset. The proposed algorithm is in the same range of values when considering the standard deviation of the accuracy, and it would be interesting to compare both their performances on another dataset.

5.7.4 Extension to the IR Spectrum

The stacked MMLBP + GLCM algorithm is first evaluated on the visible and near infrared ranges separately on the colorectal dataset. Once this done, it is evaluated on a combined dataset including both the Vis and IR data by fusing the accuracy results at a score level using the stacking technique discussed in Section 5.5.2. Table 5.7 proves that using both the visible and infrared ranges of the light spectrum improves slightly the results. On the colorectal dataset, the proposed algorithm scores 99.2 % when using only the bands representing the wavelengths in the visible spectrum; this same algorithm scores 99.5 % when using the wavelengths from the infrared as well as the visible range. One can notice that the IR alone does not perform as well as the Vis spectrum

Table 5.7: Accuracy of proposed algorithm on colorectal dataset.

Dataset	Accuracy
Dataset 3 Vis	99.2 ± 0.1
Dataset 3 IR	96.2 ± 0.5
Dataset 3 Vis+IR	99.5 ± 0.1

with this algorithm but it adds different information and helps improving the accuracy when combined.

5.8 Conclusion

Multispectral texture features form an attractive method for extracting information from histologic images of colorectal or prostate tumor tissue for classification purposes. This chapter proposed a MMLBP texture feature. The feature was combined with different classification schemes. It was proven that the feature combined with GLCM using a stacked generalisation for feature fusion at the score level for classification gives better or similar results than existing ones available in the literature. It attains a classification accuracy above 99 % on both the datasets tested. This study also showed that results can be improved when combing both infrared and visible information extracted from tissue samples.

In the next chapter, a more versatile and automatised feature extraction technique based on convolutional neural networks is introduced.

Chapter 6

Deep learning: Convolutional Neural Networks for Colorectal and Prostate Cancer Diagnosis

6.1 Introduction

Deep learning is a branch of machine learning that attempts to mimic the thinking process. In order to process data, information is passed through a network consisting of different layers, where each layer serves as input to the following layer. The first layer of a network is referred to as the input layer, while the last one is the output layer. All the layers in between are called the hidden layers. Typically, a layer is a simple algorithm consisting of an activation function.

One of the first neural networks was created in 1943 by Walter Pitts and Warren McCulloch [137]. The authors based their model on advances in the human

brain research and used a combination of algorithms called threshold logic to imitate the thinking process. In 1967, Ivankhnenko and Lapa [138] published a work describing the architecture of a deep network that had multiple thin layers of non-linear features. The principle was to select the best features using statistical methods and forward them to the next layer. The first convolutional neural networks (CNN) appeared in 1979, when Fukushima [139] designed a system called Neocognitron consisting of multiple pooling and convolutional layers. Yann LeCun demonstrated the concept of backpropagation in 1989 [140] and used it in a CNN in order to read handwritten digits.

Deep learning has kept evolving but failed to find practical application because of its processing cost and insufficiently powerful hardware technology available. In 1999, Graphic Processing Units (GPUs) were developed, making computers faster at processing data. At that time, neural networks started to compete with SVMs. They were still slower than the latter but often offered better results with the same data, with their performance improving as more data is added. In 2011, the speed of GPUs started to reach a level allowing easy CNN training and making neural networks efficient and rapid. As an example, the AlexNet architecture was developed in 2012 [141] and won several international competitions including the ImageNet competition. GoogLeNet [142] that is a 22 layers deep network, won the ImageNet competition of 2014. He *et al.* [143] deepened even more the networks with ResNet and won the best paper 2015 at the Conference on Computer Vision and Pattern Recognition. In order to reduce the training times, they developed a framework where layers are formulated as residual function with reference to the layer input, as opposed to the unreferenced learning functions previously used. Their residual network counts 152 layers. In 2016, the company Google DeepMind used a mix of supervised deep learning and reinforcement deep learning (RL) to

create a system able to learn how to play the game of Go [144]. This program called AlphaGo achieved a 99.8% winning rate against other Go programs, and defeated the human European Go champion by 5 games to 0. In 2017, they created AlphaGo Zero [145] which outperformed the original AlphaGo in performance and in learning time without using any human knowledge.

This field of machine learning is now very active and the research community is focused on solving practical applications using modern deep learning. This chapter aims at applying the deep learning framework to the problem at hand. It will first describe the principles of deep neural networks and the techniques used for optimising them. Then, the particular CNN architecture is described as well as the experiments carried out for this work. Finally, the results achieved and their analysis are detailed.

6.2 Feedforward Neural Networks

Feedforward neural networks, also called multilayer perceptrons (MLPs), are the base of deep learning models. They aim to approximate a function $f^* : \mathbf{x} \rightarrow y$, where $\mathbf{x}$ is an input feature vector and y is its corresponding class. The network builds a mapping $\mathbf{y} = f(\mathbf{x}; \theta)$ by learning the parameters θ that provide the best approximation function to f^*. In this type of networks, information moves from input to output through the intermediate layers with no feedback connections as depicted in Figure 6.1. The number of layers is called the depth of network. Each layer consists of a vector of functions or units that act in parallel and this vector's dimension is the width of the layer. Therefore, many hyperparameters need to be chosen when designing a neural network model including its architecture, that is to say the number of layers

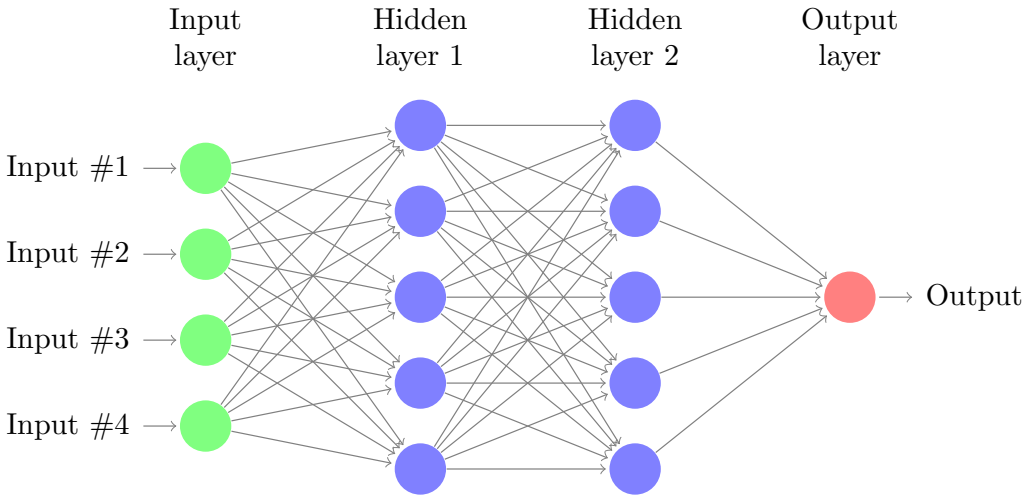


Figure 6.1: Example of a simple neural network with 2 hidden layers, each one with a width of 5.

and units per layer. Figure 6.1 shows an example of a simple architecture of a neural network.

A hidden layer computes an affine transformation of its input and then applies a non-linear function g. This is defined by $\mathbf{h} = g(\mathbf{W}^T\mathbf{x} + \mathbf{b})$, where h is the output of the hidden layer, W is the weights of the affine transformation and b the biases. W and b are the parameters learnt when training the model.

The function chosen for each unit is called the activation function and is inspired by the behaviour of biological neurons. The most widely used activation function is the Rectified Linear Unit (ReLU) defined by : $g(z) = \max\{0, z\}$. Many other options are available and the research on activation function is still a very active field but ReLU has proven to perform well and is the default choice for activation functions.

The network training is performed using a gradient descent. The main difference with other models is that the nonlinearity of neural networks causes the loss function to be nonconvex. Unlike convex optimisation used with SVMs or LR, there is no guarantee of global convergence of a gradient descent applied

to a nonconvex loss function. Consequently, the learning process is sensitive to the initial values of weights and biases. In order to apply a gradient based-learning, a cost function must be chosen. The problem at hand in this work defines a conditional distribution $p(\mathbf{y}|\mathbf{x};\theta)$ and the maximum likelihood principle is well adapted for it [34]. As a result, the cross-entropy between the training data and the model's prediction – which is equivalent to the negative log-likelihood – is used as cost function. It enables the model to estimate the conditional probability of the classes, knowing the input, and is given by:

$$J(\theta) = -\mathbb{E}_{X,Y\sim\hat{p}_{data}} \log p_{model}(\mathbf{y}|\mathbf{x}), \tag{6.1}$$

where, $\hat{p}_{data}$ is the distribution of the training data, p_{model} is the model distribution, and θ is a set of parameters for which the cost function is calculated. Consequently, the specific form of the cost function changes depending on the form of $\log p_{model}$.

6.2.1 Back-Propagation

During training, the gradient of the cost function $\nabla_\theta J(\theta)$ is computed using a back-propagation algorithm [146, 147, 148], to allow information to flow backwards through the network and compute the error made on each weight of the network. A gradient-descent is then used to minimise the cost function. Learning is subsequently performed by updating the units' weights. This procedure is detailed in Algorithm 2.

Training a neural network consists of applying a series of forward propagations – the network output is generated from the data flowing through the network – and back-propagations to compute the error at each unit. Each one of these

Algorithm 2: Back-propagation algorithm for a L-layer network with weights $\theta^{(l)}$ and a training set $\{(\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_m}, \mathbf{y_m})\}$.

for $l \leftarrow 1$ **to** L **do**
  $\theta^{(l)} =$ small random value ; `// Initialise network weights for each layer`
end
foreach *epoch* **do**
  for $l \leftarrow 1$ **to** L **do**
    $\Delta^{(l)} = 0$; `// Initialise gradient matrices`
  end
  `// For each training example`
  foreach $(\mathbf{x_i}, \mathbf{y_i}) \in \{(\mathbf{x_1}, \mathbf{y_1}), \ldots, (\mathbf{x_m}, \mathbf{y_m})\}$ **do**
    `// Forward propagation`
    $\mathbf{w}^{(1)} \leftarrow \mathbf{x_i}$;
    for $l \leftarrow 2$ **to** L **do**
      $\mathbf{w}^{(l)} \Leftarrow g(\theta^{(l-1)}\mathbf{w}^{(l-1)})$; `// For each layer of the network`
    end
    `// Back-propagation`
    $\delta^{(L)} \leftarrow \mathbf{w}^{(L)} - \mathbf{y_i}$; `// Compute the error at the output layer`
    for $l \leftarrow L - 1$ **to** 2 **do**
      $\delta^{(l)} \Leftarrow ((\theta^{(l)})^T \delta^{(l)}) .* \mathbf{w}^{(l)} .* (1 - \mathbf{w}^{(l)})$; `// Compute the error of each unit at the hidden layers`
      $\mathbf{\Delta}^{(l)} \leftarrow \mathbf{\Delta}^{(l)} + \delta^{(l)}(\mathbf{w}^{(l)})^T$; `// Update the matrix` Δ `for each layer`
    end
  end
  `// Gradient-descent: Update weights using learning rate` η `and gradient` $\frac{1}{m}\mathbf{\Delta}$
  for $l \leftarrow 1$ **to** L **do**
    $\theta^{(l)} \leftarrow \theta^{(l)} - \eta \frac{1}{m} \mathbf{\Delta}^{(l)}$
  end
end
return $\theta^{(1)}, \ldots, \theta^{(L)}$;

forward propagation and back-propagation combinations in called a pass. A pass of all the training example is performed for computing the gradient used for the gradient-descent algorithm. A pass of every training example is called an epoch. At the end of each epoch, the network's weights are updated using a learning rate hyperparameter that is multiplied to the gradient calculated with back-propagation.

The learning rate is one of the most important hyperparameters to tune in a neural network as it controls the effective capacity of the network [34]. Therefore, it needs to be carefully optimised. If the learning rate is too large, the gradient-descent can have the opposite of the desired effect and the training accuracy can decrease [149]. However, when it is too small, training is slower and sometimes the training accuracy can stay permanently small [149]. The number of epochs is also a hyperparameter to be tuned ahead of training.

6.2.2 Mini-Batch

During network training, it was previously explained that the gradient of the cost function, $\nabla_\theta J(\theta)$, was estimated as the mean of the gradients over all the training examples. However, it can be computed on a small number of examples randomly selected and averaged only over these examples.

It can be proven that the standard error of the mean estimated from n examples is given by $\frac{\sigma}{\sqrt{n}}$, where σ is the true standard deviation of the value of the samples [34]. This means that the precision gain is not linearly related to the number of examples used. The gain in precision is therefore not worth the quadratic increase of the number of examples used to estimate it. Consequently, the optimisation will converge faster if the estimates of the gradient

are computed rapidly with a greater approximation rather using an exact value that is slowly computed.

Typically, a deep learning algorithm uses a subset of the training set called mini-batch to compute the gradient estimate and perform gradient-descent. With larger batches, a more accurate estimate of the gradient is obtained but with less than linear returns. With smaller batches, the learning rate might need to be very small in order to keep the stability of the system, which could be broken by the high variance in the gradient estimate, thus increasing the computing cost. However, it was observed that small batches can provide a regularising effect [150]. Thanks to modern multicore GPUs, several examples can be processed in parallel. The runtime is consequently lower when using mini-batch training as long as the mini-batch size allows all the mini-batch examples to be processed in parallel.

6.2.3 Regularisation: Reducing Overfitting

In machine learning, the main issue is to increase the generalisation performance, even if it means a smaller training performance. Strategies designed to tackle this problem are collectively called regularisation techniques. Many different regularisation strategies are commonly used by the deep learning community and it is still an active subject of research [34].

Dataset Augmentation

The most efficient way to build a model with an improved generalisation performance is to increase the number of examples in the training set. However, in medical imaging, the size of the datasets is usually very small – 40 and

32 images for the two datasets used in this thesis. Using this amount of data would probably mean a very small test accuracy, meaning either that the model overfits the data or even a very small training accuracy due to the model underfitting the data. In order to overcome this issue, fake data is generated and added to the training set. This method has proven to be very effective for reducing overfitting [151, 152, 153, 154]. The fake data is generated using geometric transformations of the images in the dataset that does not change the class. For instance, translation, rotation, flip, skewing, rescaling or a combination of these transformations are often used. However, the transformations that can or cannot be used are specific to each classification problem as they need to preserve the image class.

Early Stopping

When training a model with a high enough capacity to overfit the data, it is often observed that the training accuracy increases over time. However, the validation accuracy typically increases at first and then starts falling after reaching a maximum. In order to have the minimum overfitting effect, the early stopping strategy is adopted and the model's parameter settings achieving the highest validation accuracy – and thus hopefully the highest generalisation accuracy – are selected. These parameters are then used for testing. Bishop [155] and Sjöberg *et al.* [156] showed that early stopping has a regularisation effect because it restricts the optimisation procedure to a small volume of the parameter space in the neighbourhood of the initial parameter value.

Dropout

The dropout was introduced in 2014 by Sriastava *et al.* [157]. It is a computationally not expensive but very effective regularisation strategy. It trains an ensemble of models consisting of networks formed by removing some units in the hidden and input layers from the original network architecture. The probability of removing each unit is a hyperparameter set per layer ahead of training. This provides a way to approximately and efficiently combine many different simulated neural network architectures sharing weights. Therefore, the number of parameters to learn during training does not change. During testing, the original network architecture without dropout is used with the scaled-down weights learnt during training of the different thinned networks. If a unit was kept with a probability p during training, the weight of this unit is multiplied by p during testing.

6.3 Deep Convolutional Networks

CNNs [140, 146] are a type of neural network that specialise in data with a grid-like topology. They are particularly adapted for image processing. Similar to conventional neural networks, they consist of units with weights and biases that are learnt during training. However, with the assumption on the data topology, it is possible to add some properties to the architecture in order to reduce the number of parameters to learn and improve the network implementation efficiency. These key ideas are: local connections, shared weights, pooling and the use of many layers [158].

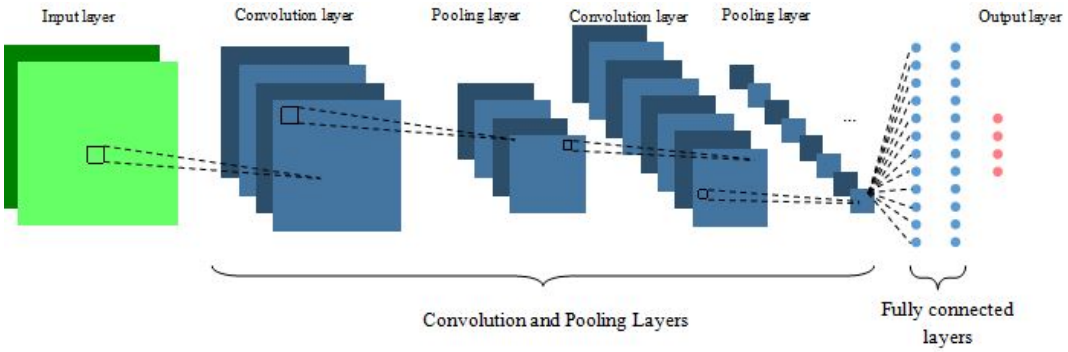


Figure 6.2: CNN architecture

CNNs' units are arranged in three dimensions in each layer of the network: width, height and depth of the activation volume. As depicted in Figure 6.2, three different kinds of layers are usually stacked to form the full CNN architecture: convolutional layer, pooling layer and fully connected layer. Fully-connected layers are layers of a traditional MLP as described in Section 6.2.

6.3.1 Convolutional Layer

The convolutional layer is the core layer of a CNN. The basic idea is that instead of connecting a unit to every unit of the previous layer, it is only connected to a local region of the previous layer. The spatial extent of this connection is called the receptive field of the unit or the filter size. It is a hyperparameter of the model. The filter size along the depth axis is the same as the depth of the previous layer. This shows an asymmetry in the way spatial dimensions (width and height) and the depth dimension are treated, making the network particularly adapted for multispectral images. The connectivity of the convolutional layer is local along the width and height but the layer is fully connected along the depth. A convolutional layer's parameters can also be seen as a set of spatially small-sized learnable filters or kernels. During the forward pass, the filters are convolved across the width and height dimensions

of the input volume. This action produces a 2D activation map outputting the responses of the filter at each position of the input layer [158, 34].

The output volume of a convolutional layer depends on three hyperparameters: the number of filters, the stride and zero-padding.

- The number of filters looking at the same receptive field determines the depth of the output volume. A different filter activates for every different pattern. A set of units with the same receptive field is called a fibre of the output layer.
- The stride, used when the filters are slid along the spatial dimensions of the previous layer, impacts the height and width of the output volume. The higher the stride, the smaller the output volume is.
- The input volume can be padded with zeros around the border in order to keep the information at the border. Without zero-padding, the information carried by the pixels at the border of the input image would vanish quickly after successive convolutional layers. This artificially increases the size of the input layer and therefore increases the size of the output layer as well.

Parameter Sharing

The parameter sharing scheme is used to reduce the number of parameters to be learnt. It is based on the assumption that a useful feature at one position of the input layer is also useful at a different position. This means that the units on a same output depth slice use the same weights and bias. This explains the fact that the forward propagation through a convolutional layer is equivalent to convoluting a filter or kernel with the input layer.

6.3.2 Pooling Layer

Typically, a pooling layer is inserted between successive convolution layers. The pooling function replaces the output of a convolutional layer at a certain unit with a statistic of its neighbouring units. The most popular pooling function used is the max pooling introduced by Zhou *et al.* [159]. The pooling layer aims at making the system invariant to small translations of the input. This property gives more importance on whether or not a feature is present in the input rather than to its exact position.

6.3.3 CNN, Feature Extraction and Classification

The combination of convolutional and pooling layers aims at learning the best features that could be extracted from the dataset. It contrasts with most of the current methods that use handcrafted feature extraction techniques such as the ones presented in the previous chapters. These approaches can give very good results but are usually sensitive to the dataset and perform poorly when applied to different data. The combination of convolutional and pooling layers of a CNN provides a more versatile way to extract features from images. The fully-connected layers of a CNN correspond to the classifier. It aims at learning to classify the learnt features. As a result, a CNN is a unified versatile scheme for feature extraction and classification. Because medical images classification is often a very complex task, it requires carefully manufactured feature sets for each type of data or even each different dataset. Doing just that with a unified framework, CNNs seem particularly adapted to the field.

6.4 Experiments

6.4.1 Hardware and Software Specifications

In order to train deep CNNs, a GPU is needed. The system used for this experiment is equipped with one NVIDIA K80 GPU and four CPUs. It has 61 GiB RAM. Regarding software, Keras with TensorFlow backend was used. Keras has the advantage of making available deep learning models alongside pre-trained weigths.

6.4.2 Selected Architecture

The proposed CNN architecture evaluated for the task at hand was based on VGG16 [160] whose architecture is represented in Figure 6.3. In oder to design the proposed architecture, the last block of convolutional layers of the VGG16 was removed and the number of filters per layer was halved. The idea is to reduce the capacity of the network because the inter-class similarity in the datasets used for the task at hand is high compared to the dataset on which VGG was tested.

As represented in Figure A.1 and A.2, the overall proposed network architecture consists of a total of 13 layers with weights, the first ten being convolutional layers and the remaining three fully-connected. The output of the last fully-connected layer is fed to a softmax classifier, which is a generalisation of the LR classifier to the multiclass problem and produces a distribution over the four class-labels. The network uses the cross-entropy as loss function.

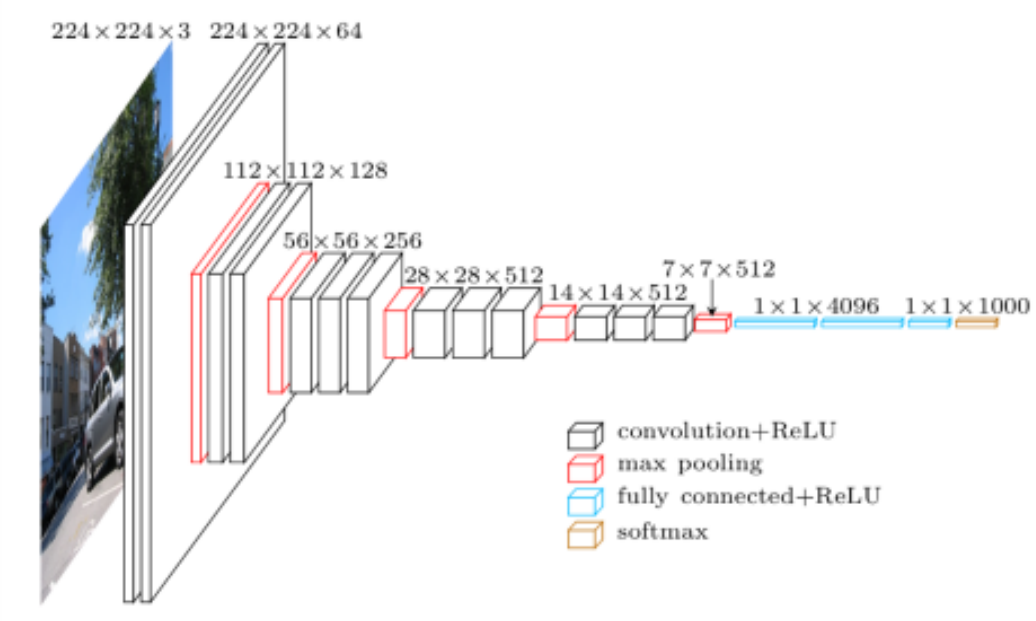


Figure 6.3: Illustration of the architecture of VGG16 [160]

Like in VGG16, it is decided to use a small kernel with a size of 3 pixels for every convolutional layer. The strategy of stacking convolutional layers with small filter size is preferred to the one using a single large receptive field convolutional layer. For the same final receptive field, the former strategy includes non-linearities (ReLU functions) at each layer while the latter computes a simple linear function on the input which makes the features less expressive. A stride of 1 is also adopted for the whole network in order to minimise information loss.

In order to have a better control over the outputs size of each layer and keep border information, a zero-padding of one is added before each convolutional layer. The first two convolutional layers use 32 kernels and are followed by a 2×2 max-pooling layer as described in Section 6.3.2. The max-pooling layer reduces the size of the ouput and thus the network capacity. In order to compensate this loss, the number of kernels is doubled in the next convolutional layer. Consequently, this sequence is followed by two convolutional layers with

64 filters, and then a new max-pooling layer is applied. This is then followed by a series of three convolutional layers with 128 filters and a max-pooling layer. A final series of three convolutional layers with 256 filters and a max-pooling layer is applied. The neurons in the three fully-connected layers with sizes 1024, 1024 and 4, respectively, are connected to all neurons in the previous layer. The ReLU non-linearity is applied to the output every layer with weights.

Dropout is used after every max-pooling and fully-connected layer to reduce overfitting. An early stopping strategy is also adopted in order to reduce the training time and for regularisation reasons, as explained in Section 6.2.3. Finally, a data augmentation is carried out using the following transformations: each image is flipped along the two spacial axis and a 30° rotation in both directions is applied. This results in the generation of 27 fake images for each real data image. To ensure that the generalisation is not over estimated, the dataset augmentation is performed after splitting the dataset into training and test sets.

6.4.3 Details of Learning

The weights of each layer are initialised using a Xavier initialisation method [161], where the weights are drawn from a normal distribution centered on zero and with a standard deviation of $\sqrt{\frac{2}{N_{in}+N_{out}}}$, where N_{in} and N_{out} are the number of input and output units, respectively. The network is trained separately on the two datasets.

The learning rate used is the same for all the layers. It is optimised using a grid-search scheme which results are presented in Figure 6.4 and 6.5. The accuracy is computed for different learning rates taken from a logarithmic scale

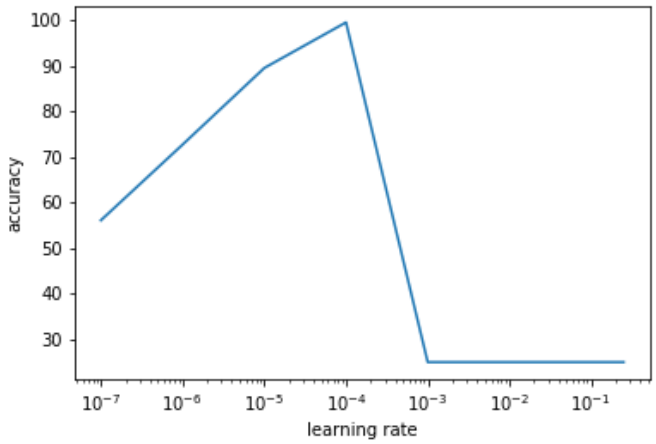


Figure 6.4: Validation accuracy obtained with different learning rates for the network trained on prostate data.

and one value per decade is evaluated. The learning rate selected for training is then 0.0001 for both datasets.

For each model training, a 10-fold cross-validation technique is adopted in order to find a good estimate of the systems' generalisation accuracy. This provides a large training set for better learning.

Figure 6.6 and 6.8 illustrate the evolution of the loss function during training for the prostate and colorectal datasets, respectively. Figure 6.7 and 6.9 show the evolution of their accuracy. It can be noticed from these figures that the validation accuracy is very close to the training accuracy which proves that the model is not in the overfitting regime. The higher variation in validation accuracy and loss can be explained by the smaller set used for validation compared to the one used for training.

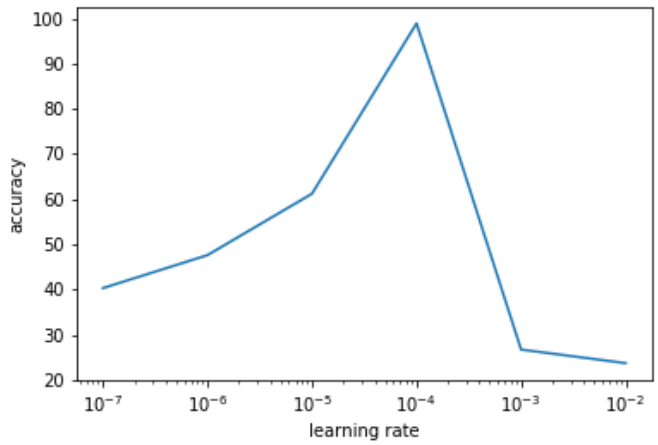


Figure 6.5: Validation accuracy obtained with different learning rates for the network trained on colorectal data.

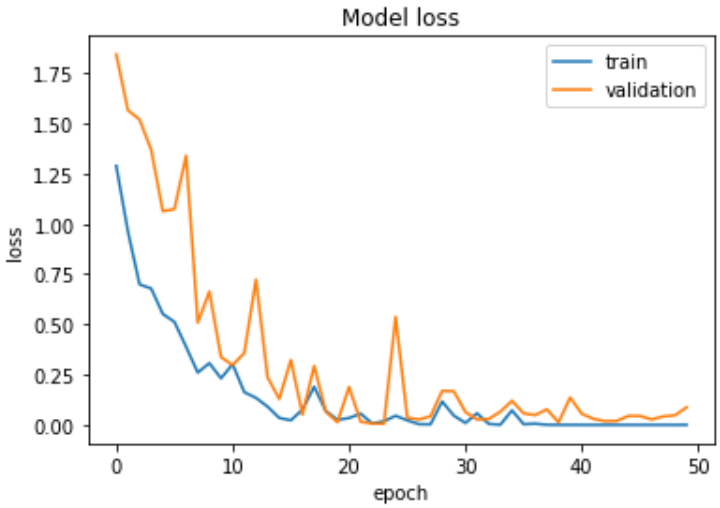


Figure 6.6: Loss function evolution during training for the prostate dataset

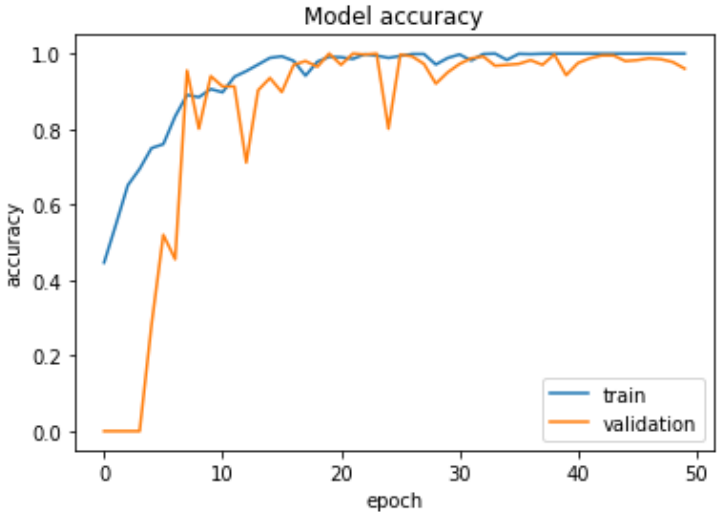


Figure 6.7: Accuracy evolution during training for the prostate dataset

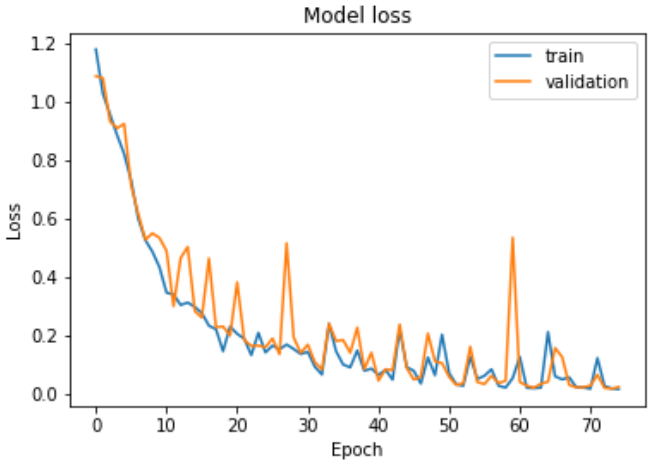


Figure 6.8: Loss function evolution during training for the colorectal dataset

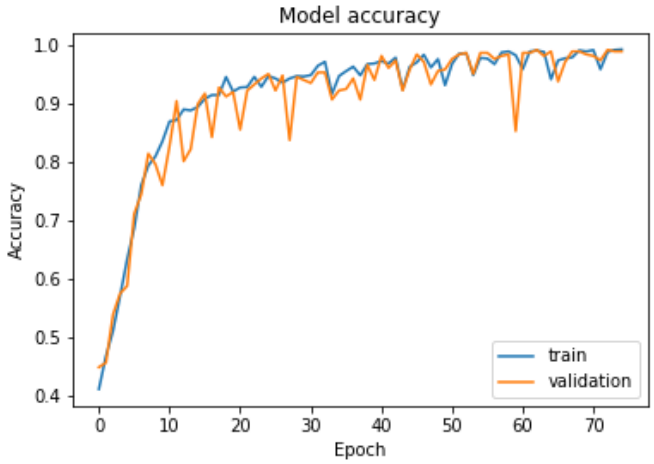


Figure 6.9: Accuracy evolution during training for the colorectal dataset

6.4.4 Transfer Learning

Transfer learning consists of using a network previously trained on another dataset in order to use the knowledge acquired during this learning task for the new task at hand [162]. In most transfer learning for image classification tasks, the ImageNet dataset [163], which contains 1.2 million images with 1000 categories, is used for pre-training the network. When only a small dataset is available, this allows to train the CNN on a very large dataset and therefore train a high capacity network that captures fine details without overfitting. Very deep networks also require a lot of time and very powerful machines equipped with multiple GPUs. Using pre-trained networks can be advantageous when not provided with appropriate resources. Several transfer learning scenarios are practicable.

In a first scenario, the pre-trained CNN is used as a fixed feature extractor. The convolutional layers of the network are kept with the weights determined during training on the ImageNet dataset and the pre-trained fully-connected layers are replaced with fully-connected layers initialised with random weights.

During training, only the newly added fully-connected layers are marked as trainable. They use the features extracted by the pre-trained convolutional layers as inputs. These features are usually referred to as CNN codes [34, 162].

Another strategy is, on top of retraining the fully-connected layers from scratch, to fine-tune the weights of the pre-trained convolutional layers by continuing back-propagation. Either all the convolutional layers can be fine-tuned or only some of the higher-level layers to avoid overfitting. This derives from the observation that the lower-level layers usually learn more generic features, such as edge detectors, that can be used for many different learning tasks. On the other hand, the high-level layers tend to learn features that are more and more specific to the characteristics of the classes of the original dataset.

In this thesis, only the first scenario has been investigated. The pre-trained CNNs are very deep and require a very high computation power to be fine-tuned. Using them as feature extractors is in fact equivalent to only training a relatively shallow MLP.

The proposed architecture was compared to popular CNN architectures: VGG16 [160], InceptionV3 [142], ResNet50 [143]. These networks were initialised with the weights obtained when pre-training them on the ImageNet dataset. However, InceptionV3 and ResNet50 being very deep networks (48 and 152 layers, respectively), a minimum input image size is required. InceptionV3 necessitates a minimum width and height of 139 pixels and ResNet50 of 197 pixels. The images of the colorectal dataset being smaller, a zero-padding was added to reach the required dimensions. Moreover, the ImageNet images are RGB images and therefore have a depth of 3 channels. In order to meet the dimension requirements, a PCA was carried out to reduce the dimensionality of the multiscale images to 3 channels.

Table 6.1: Validation and test accuracy comparison of different architectures (in %)

Method	Prostate dataset		Colorectal dataset	
	Validation accuracy	Test accuracy	Validation accuracy	Test accuracy
Proposed CNN	100	99.8 ± 0.1	100	99.5 ± 0.1
VGG16 Xavier initial.	100	99.6 ± 0.1	99.0 ± 0.1	99.2 ± 0.1
VGG16 pre-trained	100	99.5 ± 0.1	97.5 ± 0.2	98.1 ± 0.1
InceptionV3 pre-train.	98.8 ± 0.2	99.0 ± 0.1	92.3 ± 0.3	94.5 ± 0.3
ResNet50 pre-trained	100	100	99.5 ± 0.1	99.0 ± 0.2

6.5 Results and Analysis

In order to visualise the effect of the kernels on images throught the network, Figure 6.10 and 6.12 present examples of outputs of the first convolutional layer of the networks trained with the prostate and colorectal datasets, respectively. Figure 6.11 and 6.13 depict examples of outputs of the last convolutional layers of the same networks. It can be observed that after the first layer, the outputs are very similar to the input image, for instance with transformations resembling edge detections. Once the image has flown through the network, different regions or features of the input image are represented in the outputs of the last convolutional layer. The different layers thus learn a succession of transformations leading to an isolation of relevant regions or features of the input image. The fully-connected layers of the network are then able to classify these particular features into the four classes.

Table 6.1 displays the validation and test accuracies obtained with the prostate and colorectal datasets for different CNN models. The evolutions of loss and accuracy during training for each model are displayed in Section A.2.

Table 6.1 shows that the validation and test accuracies are very close, proving a good generalisation of the systems and that overfitting was avoided. The

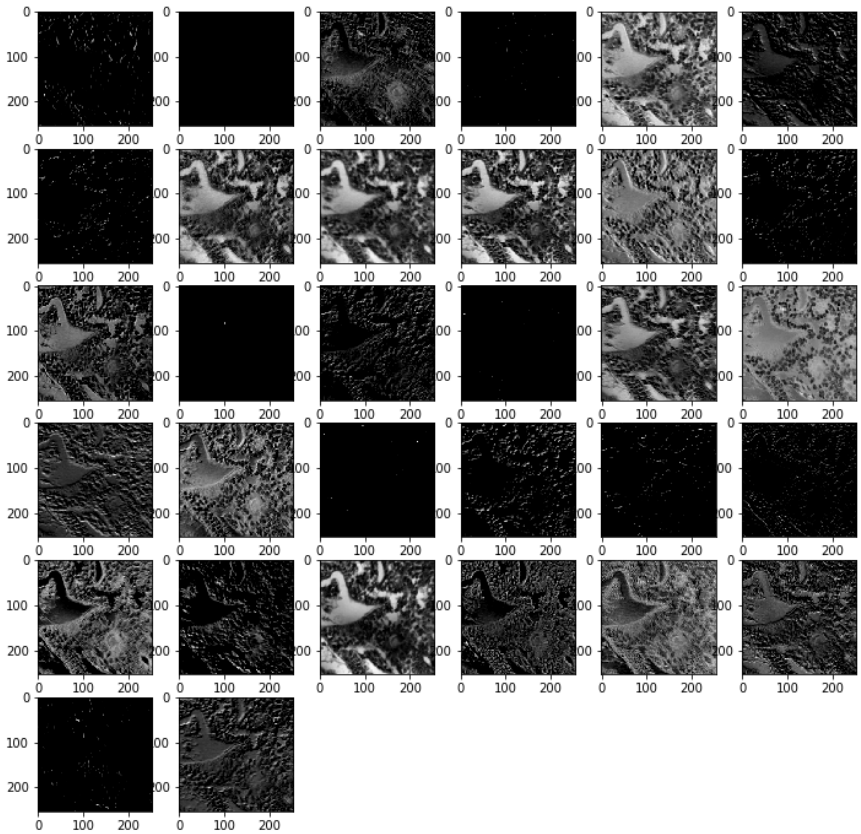


Figure 6.10: Example of an output of the first convolutional layer for the network trained on the prostate dataset

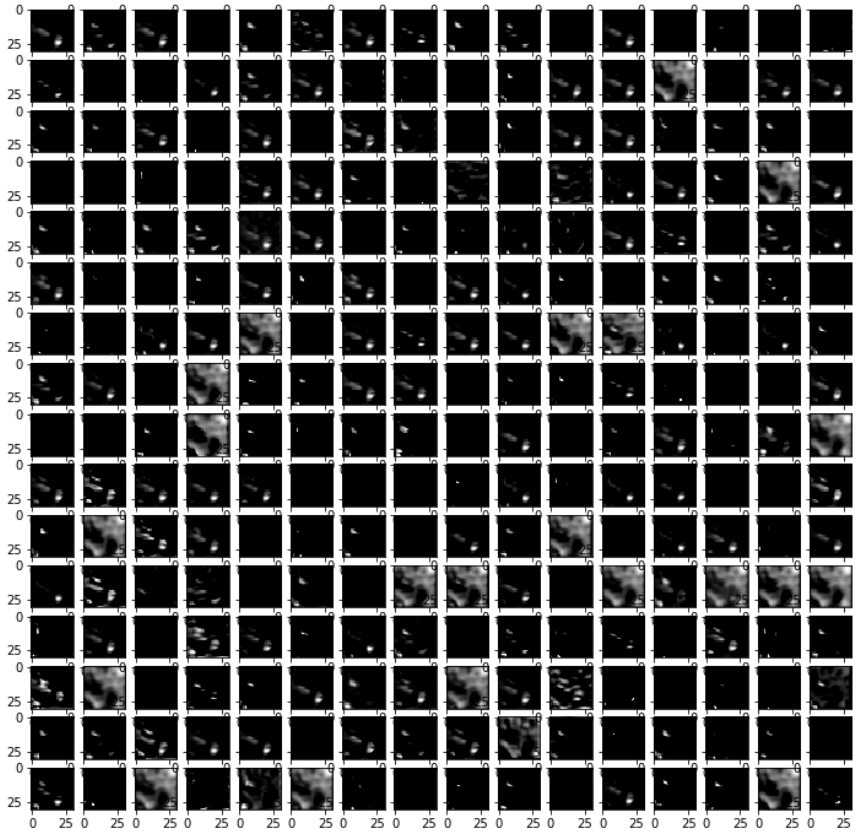


Figure 6.11: Example of an output of the last convolutional layer for the network trained on the prostate dataset

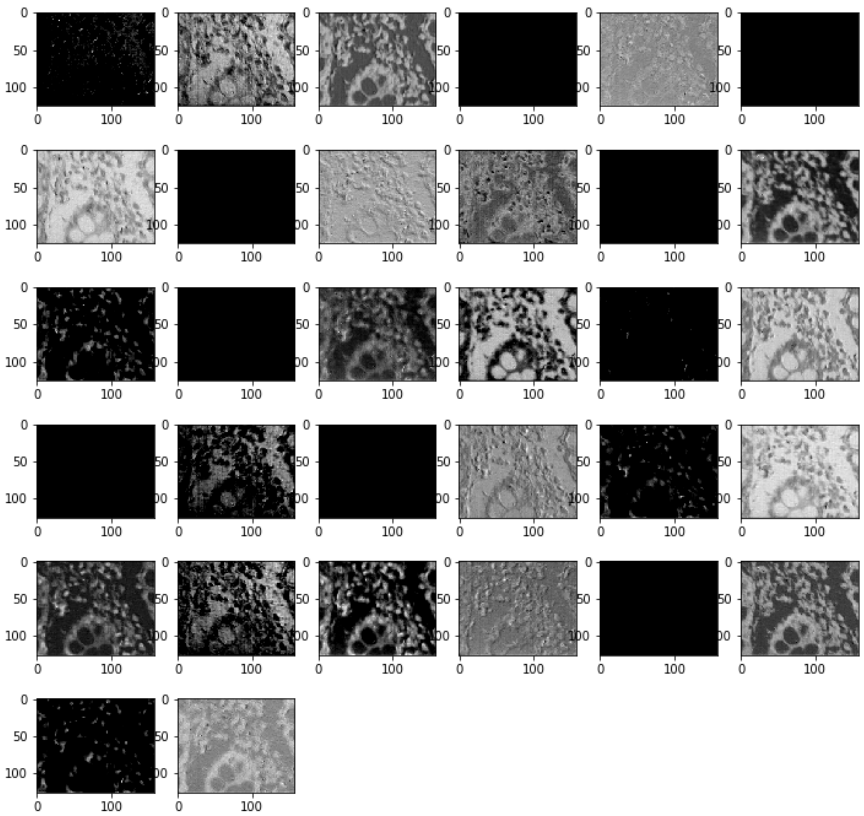


Figure 6.12: Example of an output of the first convolutional layer for the network trained on the colorectal dataset

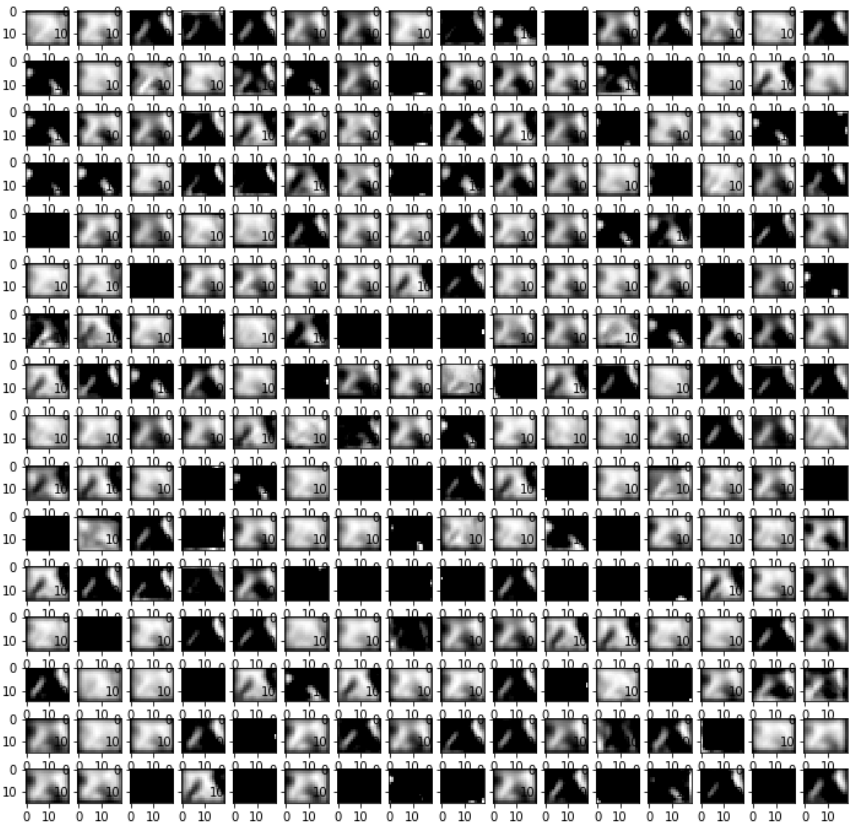


Figure 6.13: Example of an output of the last convolutional layer for the network trained on the colorectal dataset

proposed CNN model achieves an average test accuracy of 99.8 % and 99.5 % for the prostate and colorectal datasets, respectively. Figure 6.6 and 6.8 show that the optimal CNN weights were obtained after 44 and 70 epochs, respectively. The VGG16 model initialised with Xavier weights trains very "quickly" for the prostate dataset – optimal validation accuracy is obtained after 19 epochs as illustrated by Figure A.3. However, it is less efficient at learning for the colorectal dataset and needs as much as 70 epochs to obtain minimum validation loss (Figure A.5). The results also show a slight overfitting for the colorectal dataset, as the validation accuracy is lower than the training accuracy (Figure A.5). This is due to the high capacity of the network. When using this network with pre-trained weights from ImageNet, the training loss reaches a minimum after only a few epochs but the validation loss shows that the network overfits marginally for both datasets (Figure A.7 and A.9). The test accuracy is also lower than for the proposed CNN with 99.5 % and 98.1 %, respectively. This is because the CNN codes learnt with the ImageNet dataset are not as adapted to the classification task at hand than the ones learnt with the proposed CNN. The InceptionV3 model shows a higher overfitting (Figure A.11 and A.13) and a lower generalisation for both datasets with 99.0 % and 94.5 % accuracy for the prostate and colorectal dataset, respectively. This shows once again that the CNN codes learnt on the ImageNet dataset with this network are not adapted to the classification task at hand. Finally, the pre-trained ResNet50 achieves optimal accuracy with the lowest number of epochs: 5 and 22 for the prostate and colorectal dataset, respectively. It also achieves 100 % average accuracy for the prostate dataset, outperforming the proposed CNN and 99.0 % for the colorectal dataset, which is slightly lower than the proposed dataset. This lower performance compared to the proposed CNN architecture for the colorectal dataset might be due to some loss of information

Table 6.2: Accuracy comparison against other methods (in %)

Method	Prostate dataset	Colorectal dataset
Tahir *et al.* [8]	98.9	N/A
Bouatemane *et al.* [21]	99.83	N/A
Concatenated LBP [2]	92.4 ± 0.4	88.2 ± 0.5
Stacked MMLBP + GLCM [1]	99.5 ± 0.3	99.5 ± 0.1
Proposed CNN	99.8 ± 0.1	99.5 ± 0.1
ResNet50 pre-trained	100	99.0 ± 0.2

when performing PCA on the 42 channels of the colorectal dataset images. The prostate dataset consisting of images with only 16 channels, it is logical that the loss of information is not as important during this transformation.

Therefore, the proposed CNN architecture is more adapted to the task at hand than the other methods it was compared to. However, ResNet50 shows a very good performance when used as a feature extractor and is trained with fewer epochs needed.

In every case, it can be noted that the colorectal dataset is more prone to overfitting. This is probably due to the size of the images, which are spatially smaller than for the prostate dataset. As a consequence, a model with the correct capacity for the prostate dataset might be over-dimensioned for the colorectal dataset.

Comparison Against Other Machine Learning Methods

Table 6.2 shows the test accuracy of the best performing CNN architectures compared to other methods from Tahir *et al.* [8], Bouatemane *et al.* [21] and from the systems presented in Chapter 4 and 5. Regarding the prostate dataset, four systems have an accuracy above 99 %: Bouatemane *et al.* [21], Stacked MMLBP + GLCM, the proposed CNN and ResNet50 with pre-trained

weights. The highest classification accuracy is achieved by ResNet50 with 100 %. The proposed CNN and Bouatemane *et al.* [21] both achieve 99.8 %, however, the standard deviation is not given for the latter. Therefore, it is not possible to know the precision of this accuracy estimation. The Stacked MMLBP + GLCM system achieves 99.5 % with a 0.3 pp standard deviation, as presented in Chapter 5, which makes this performance similar to the proposed CNN. However, the higher standard deviation shows a lower precision on the accuracy estimation. The proposed CNN is therefore preferred.

With respect to the colorectal dataset, only the algorithms presented in this thesis were analysed for comparison. The Stacked MMLBP + GLCM system and the proposed CNN both give the same accuracy and standard deviation. They outperform the ResNet50 with pre-trained weights by 0.5 pp.

Finally, when considering the results obtained with both datasets, the Stacked MMLBP + GLCM system and the proposed CNN appear to give the most stable results as well as the highest accuracy. Yet, on average, the standard deviation of the accuracy achieved by the proposed CNN is lower than the one obtained with the Stacked MMLBP + GLCM system. The ResNet50 network's performance seems to be more dependent on the dataset used. Moreover, it would be interesting to compare the system proposed by Bouatemane *et al.* [21] using the colorectal dataset in order to verify whether it performs as well on different datasets. Considering the current information available on the systems performance and with the datasets available, the proposed CNN is selected as the best performing system in terms of accuracy for the classification task at hand.

Table 6.3: CNNs average classification computation times for one image

Method	Prostate dataset	Colorectal dataset
Proposed CNN	14 ms	7 ms
VGG16 Xavier initial.	75 ms	42 ms
VGG16 pre-trained	75 ms	42 ms
InceptionV3 pre-train.	63 ms	42 ms
ResNet50 pre-trained	65 ms	47 ms

Computational Complexity Analysis

In CADSs, an unlabeled image is fed to a previously trained system. Consequently, the time used to process this image is decisive, as it is crucial that the CADS works on-line. However, a forward pass of an image through the CNN architectures studied in this thesis is computationally non-expensive. Table 6.3 displays the classification times per image for all the CNN architectures tested. It demonstrates that only a few milliseconds are needed to classify one image, once the CNN has been trained. However, it must be noticed that the proposed CNN architecture is much quicker at classifying the images than the others. This is due to the fact that, for the architectures described in the literature and the pre-trained networks, a PCA must be carried out in order to reduce to 3 the number of channels of the image to be classified. This preprocessing stage lengthens the total classification time.

As said above, the training is performed only once when the CADS is created. Consequently, the training time is not a critical measure for the problem at hand. However, the computational complexity of deep learning systems can rapidly become significantly high. Such architectures require high-performing hardware, including GPUs. Some extremely deep architectures can also entail several weeks of training times [34]. Such long training times considerably slow down the CADS development process. In order to verify that the proposed

Table 6.4: CNNs average training computation times for the complete dataset (in s)

Method	Prostate dataset		Colorectal dataset	
	Time per epoch	Total training time	Time per epoch	Total training time
Proposed CNN	90	3780	45	2925
VGG16 Xavier initial.	245	4655	97	6790
VGG16 pre-trained	83	3154	35	1400
InceptionV3 pre-train.	39	1755	15	705
ResNet50 pre-trained	41	205	32	704

system can be trained within a reasonable duration, a comparison of the training times for each architecture is carried out (Table 6.4). The computational times depending on the hardware and software used, it is not possible to compare the CNN architectures against the other classification systems presented in this thesis. However, this is the first time deep learning is used for this application. Therefore, this section aims to establish the ability of deep learning systems to be trained in a short period of time with the datasets used.

Unsurprisingly, Table 6.4 demontrates that pre-trained networks have a much shorter training time per epoch due to the reduced number of layers to be trained: ResNet50 and InceptionV3 only train in a few minutes. When considering this measure of performance, the best architecture is ResNet50. However, the total training time for every CNN model is under two hours, making it a reasonable time for developing a CADS.

6.6 Conclusion

This chapter explained the theory behind deep learning by presenting deep feedforward networks and CNN architectures. Then, a proposed CNN archi-

tecture was detailed and compared against previously trained network models used as feature extractors. These CNNs were also compared to other classification methods presented on other chapters of this thesis and other works from the literature. The proposed CNN demonstrated excellent performances when compared to pre-trained CNNs and to the other classification methods studied in this thesis. The computational complexity of the CNNs was also analysed and it was demonstrated that the proposed CNN is faster at classifying images than the pre-trained networks because it avoids a preprocessing phase. The conclusion of this overall analysis was that the proposed CNN architecture was globally the best performing system for classifying colorectal and prostate tumour images.

Chapter 7

Conclusion

7.1 Introduction

This chapter presents a review of the main contributions made to automatic classification of microscopic images of colorectal and prostate tumours. Suggestions for future works are subsequently examined.

7.2 Summary of Thesis Contributions

CAD is a very active field of research. Many different methods have been investigated for automated classification of tumour biopsies in the past few decades. However, these systems' accuracy still needs to be improved before clinical use. This thesis aims at building a system which further improves the performance of sample classification. For this purpose, two different datasets were used to carry out experiments. The first dataset included four classes of prostate tumour microscopic images. Another dataset consisted of four classes

of colorectal tumour microscopic images. As presented in chapter 2, these organs have similar types of tissue and consequently develop the same kind of tumours. The observations made by pathologists for diagnosis are very similar in both cases. It is therefore understandable that similar automatic systems can accomplish comparable performances for both tissue types. Non-binary datasets are used with the aim of simulating a representation of the evolution spectrum of cancer, and tracking its evolution from healthy tissue, to cancer, via pre-cancerous tumours. This study exploits multispectral imagery in order to profit from the complete spectral range of the tissue's reflected light, and increase the amount of information acquired. For this work, texture features were chosen for their remarkable discrimination power, which is evident even when the tissue structure is largely altered by advanced stages of cancer. Multiple classification techniques were investigated in this thesis and the analysis of the results led to the following contributions to knowledge:

- A multiclass classification system adapted to multispectral prostate and colorectal tumour images was proposed. This system uses a two-dimensional texture extraction technique on each spectral band. The image descriptor is a concatenation of the texture feature vectors from each band, followed by a feature selection method in order to avoid problems caused by the curse of dimensionality. From the analysis of the system's performance on panchromatic and multispectral images, it was deduced that multispectral data led to a considerably higher classification accuracy than the one found with panchromatic images.

- A novel multispectral texture feature, referred to as MMLBP, was proposed. It is based on LBP features and exploits the inter-band spectral information by expanding the pixel's neighbourhood considered in LBP

patterns to the spectral dimension. Using this technique, it is possible to make more efficient use of the spectral information as opposed to concatenating the texture from each spectral band. It also keeps the feature vector to a small number of dimensions. The classification results of this method demonstrated that the proposed feature outperforms standard texture extraction methods.

- Complex classification schemes were investigated in order to improve the results achieved with the proposed MMLBP feature. The BoF framework, inspired from text classification, computes the texture of image sub-blocks and constructs a histogram of the sub-blocks feature used as image descriptor. The stacked generalisation framework uses multiple classifiers, each fed with a feature vector from a different scale. The ultimate classification decision is made by a meta-classifier that takes as input the outputs of the different classifiers. The BoF scheme is better at capturing local information, while the stacked generalisation scheme is better at selecting the features with the most discriminative power. As a result, both methods helped to improve the performance of the MMLBP texture feature.

- The colorectal dataset was acquired with a light spectral range extended from the visible wavelengths to the IR. The performance of the proposed classification system were evaluated on the visible end of the light spectrum and compared to the system's accuracy when the IR spectral bands were added to the input images. This demonstrated that including the IR information improves the classification accuracy.

- Different deep learning architectures were investigated. A CNN architecture was proposed and the results of the classification were compared to

the performance of pre-trained networks. The transfer learning systems, using features learnt from images of a different origin, showed promising results. Howerver, they did not demonstrate the same consistancy when compared to the proposed architecture with randomly initialised weights.

7.3 Future Work

From the experimental work carried out in this thesis, several directions for further research arise.

- The techniques studied in this thesis are only one part of a complete CADS. Due to the type of images available, they focus on the classification of tumours, the datasets having been constructed with images of homogeneous diagnosis regions of the biopsies. A complete CADS would take as input an image of the entire biopsy, possibly including regions with different diagnoses. Consequently, a phase of segmentation needs to be combined to this system in order to distinguish between regions corresponding to different diagnoses.

- The main problem faced by research on automated diagnosis systems is lack of data. As explained in this thesis, every research group uses different datasets for prostate and colorectal tumour classification. Moreover, each dataset consists of a limited amount of images. One of the conclusions of this thesis is that CNNs seem particularly promising for extracting the best features resulting in an excellent classification performance. However, deep learning needs an extensive amount of data in

order to be trained. It would therefore be best to generate a reference large open-source dataset consisting of several thousands of multispectral images divided into many classes, to simulate the evolution of tumours and cancers. Having a universal dataset which all systems use would help to identify the most efficient system and would also help developing deep learning systems.

- Generating a very large dataset of labeled data poses some problems. First, it is a time and labour-intensive task which requires multiple highly experienced pathologists to carry out. The main issue, however, is the element of human error, meaning that pathologists may disagree on a diagnosis for some samples. The subjectivity involved in data labeling cannot be completely removed from the process as the supervised learning system will be trained on a dataset generated by pathologists. Many deep learning systems have been designed to tackle unsupervised learning, however, none has succeeded in solving the problem in the same way that deep learning has done for supervised learning. A great challenge would be to design an unsupervised learning system that would automatically distinguish between similar types of tumour or stages of cancer. Such a system might be able to pick up on earlier stages of cancer by recognising structures invisible to the human vision system.

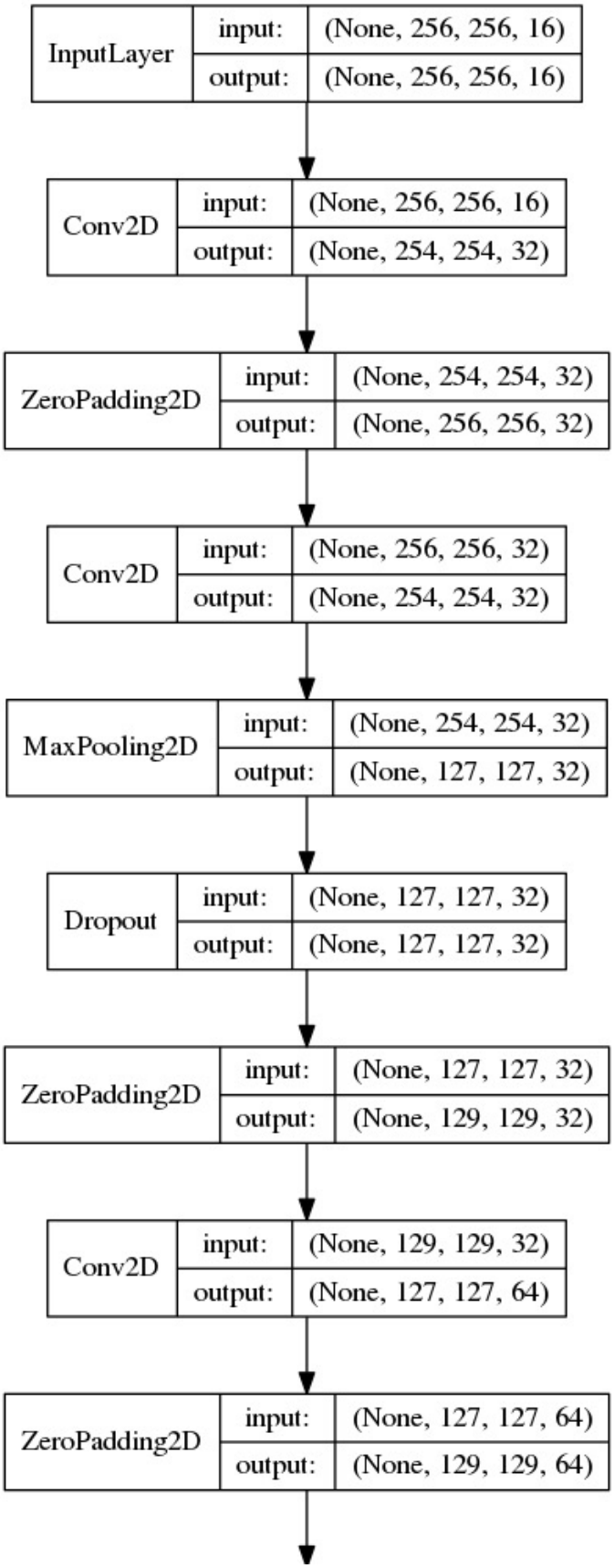
- The proposed MMLBP has the ability to characterise multispectral texture as demonstrated by this study. It can be applied to many other fields where multispectral data are involved, such as facial recognition or satellite imagery. This texture feature can help to detect objects or segment regions of interest in several applications.

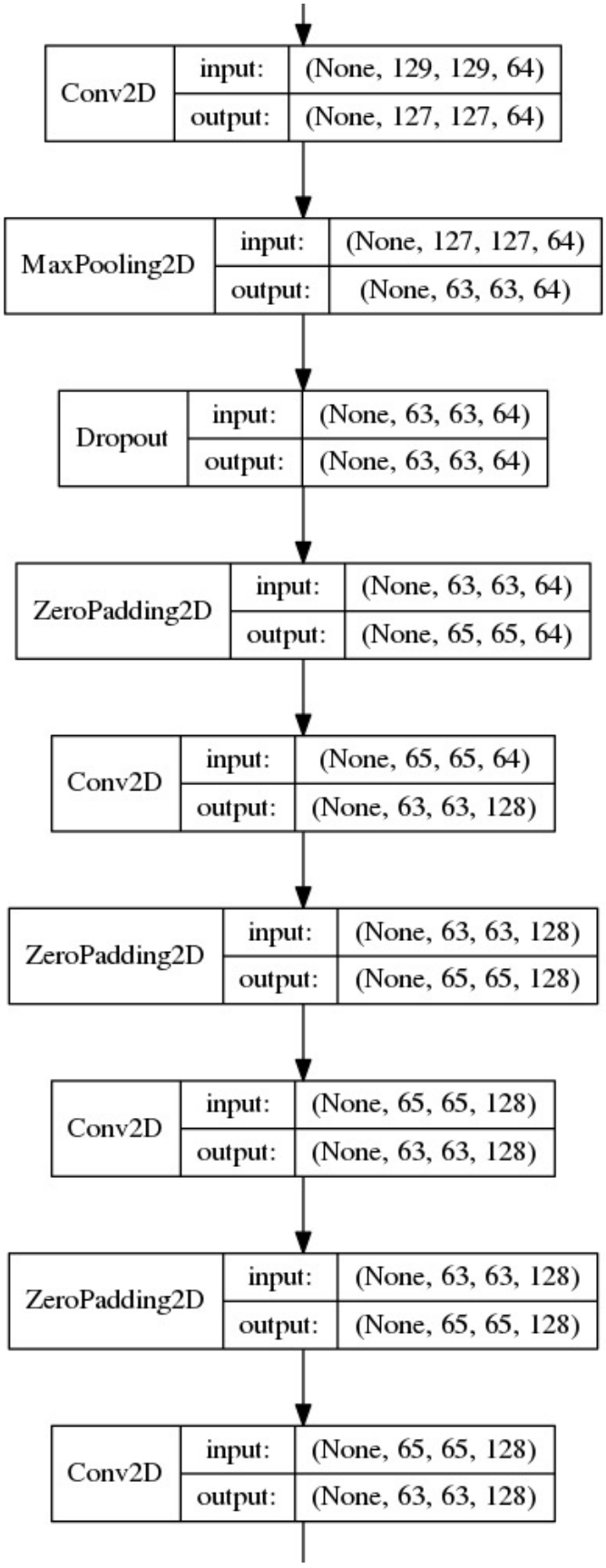
Appendix A

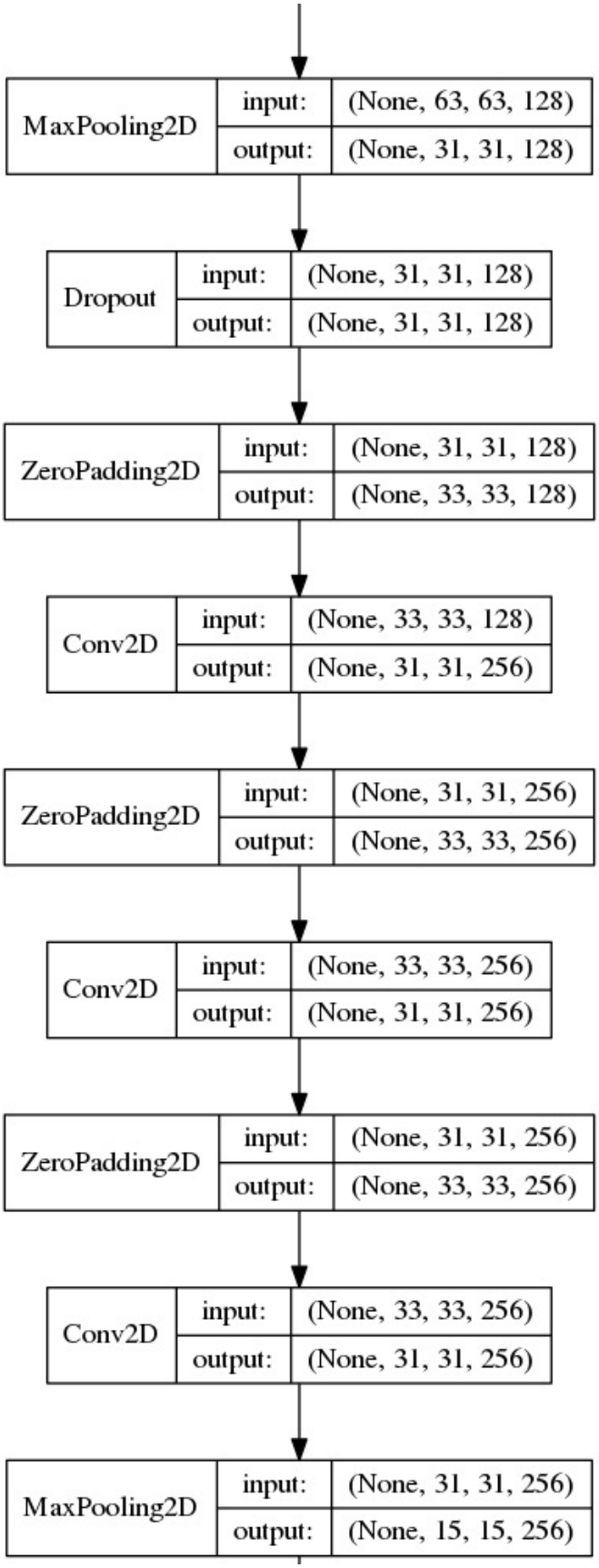
Appendices

A.1 Model Architecture of the Proposed Convolutional Neural Network

Figure A.1: Convolutional Neural Network architecture for the prostate dataset




Conv2D
input: (None, 129, 129, 64)
output: (None, 127, 127, 64)
MaxPooling2D
input: (None, 127, 127, 64)
output: (None, 63, 63, 64)
Dropout
input: (None, 63, 63, 64)
output: (None, 63, 63, 64)
ZeroPadding2D
input: (None, 63, 63, 64)
output: (None, 65, 65, 64)
Conv2D
input: (None, 65, 65, 64)
output: (None, 63, 63, 128)
ZeroPadding2D
input: (None, 63, 63, 128)
output: (None, 65, 65, 128)
Conv2D
input: (None, 65, 65, 128)
output: (None, 63, 63, 128)
ZeroPadding2D
input: (None, 63, 63, 128)
output: (None, 65, 65, 128)
Conv2D
input: (None, 65, 65, 128)
output: (None, 63, 63, 128)


MaxPooling2D
input: (None, 63, 63, 128)
output: (None, 31, 31, 128)
Dropout
input: (None, 31, 31, 128)
output: (None, 31, 31, 128)
ZeroPadding2D
input: (None, 31, 31, 128)
output: (None, 33, 33, 128)
Conv2D
input: (None, 33, 33, 128)
output: (None, 31, 31, 256)
ZeroPadding2D
input: (None, 31, 31, 256)
output: (None, 33, 33, 256)
Conv2D
input: (None, 33, 33, 256)
output: (None, 31, 31, 256)
ZeroPadding2D
input: (None, 31, 31, 256)
output: (None, 33, 33, 256)
Conv2D
input: (None, 33, 33, 256)
output: (None, 31, 31, 256)
MaxPooling2D
input: (None, 31, 31, 256)
output: (None, 15, 15, 256)

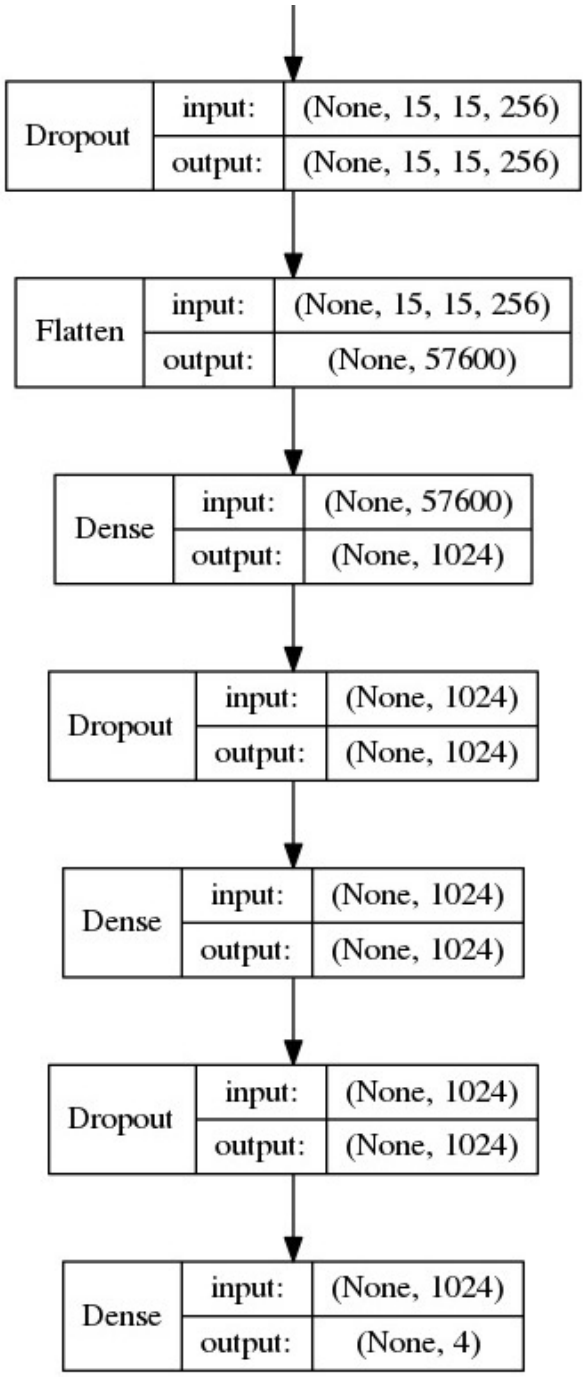
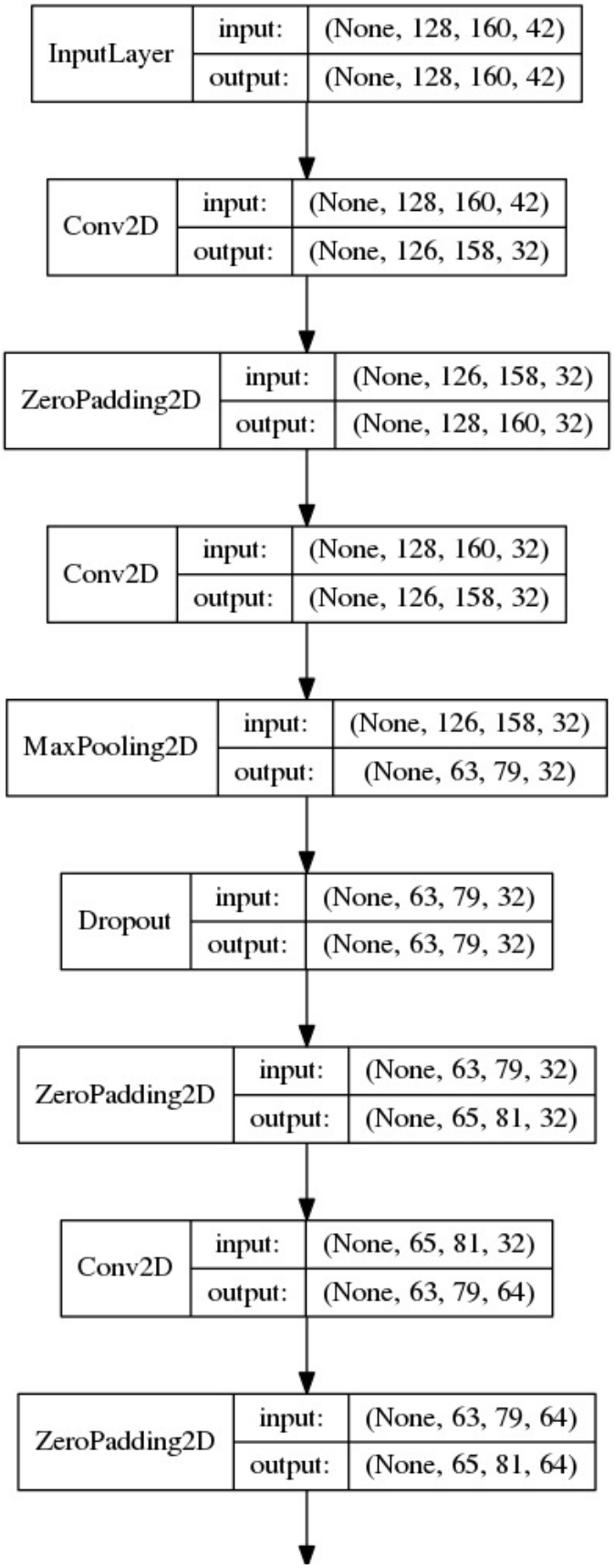
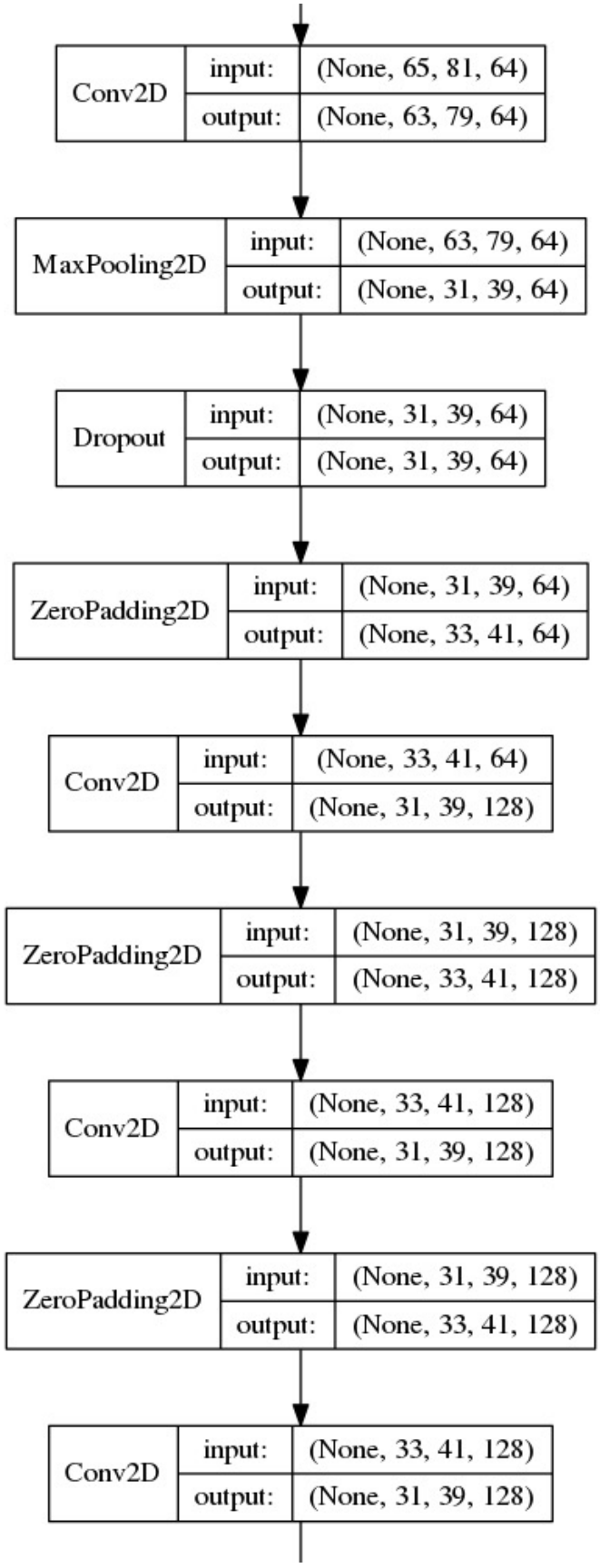
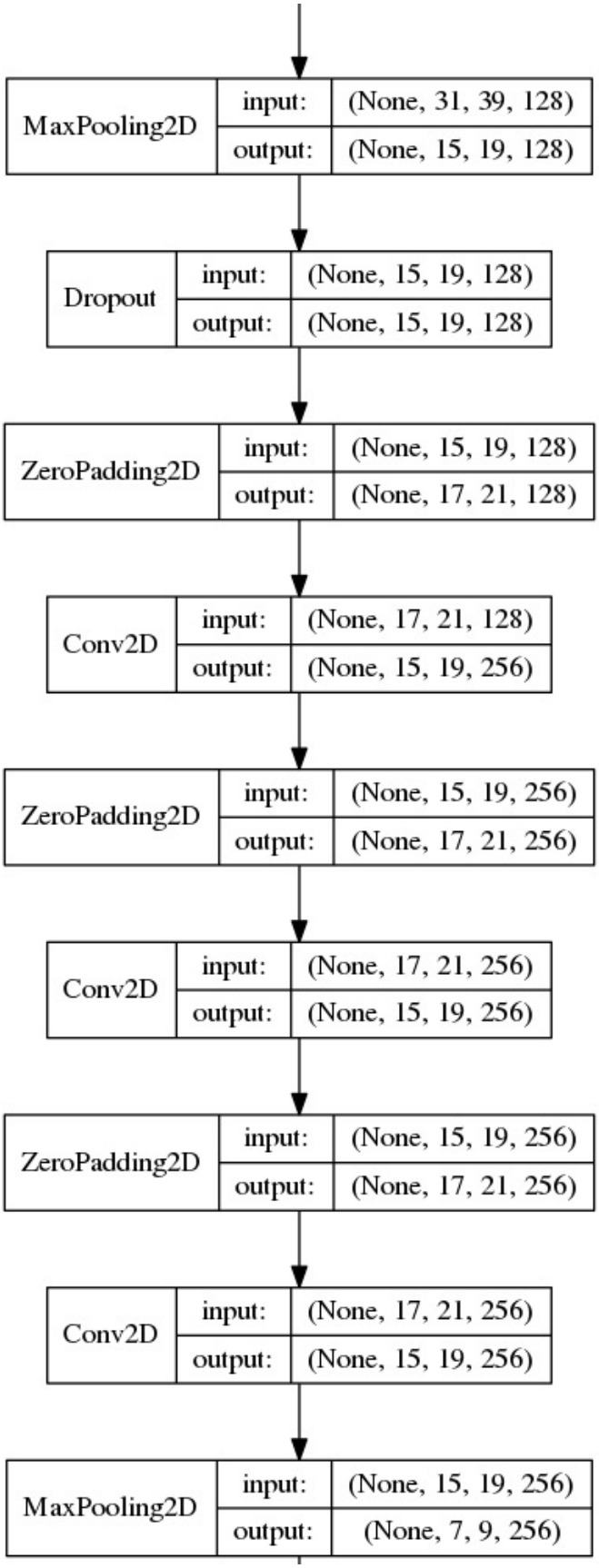
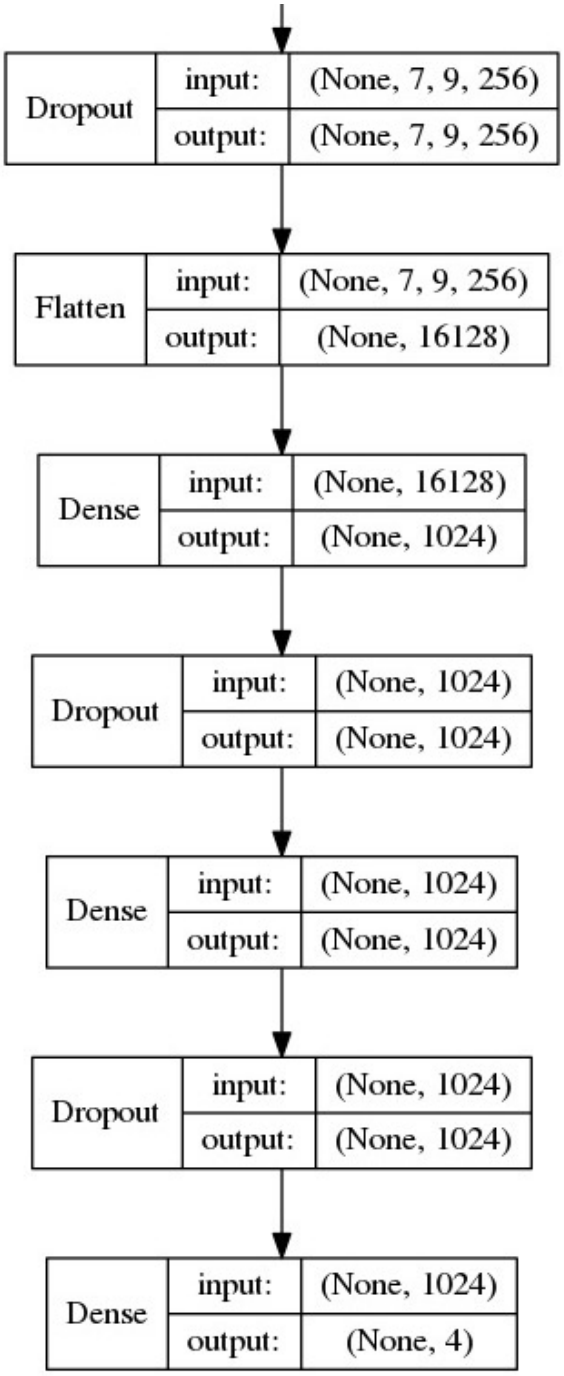

Dropout
input: (None, 15, 15, 256)
output: (None, 15, 15, 256)
Flatten
input: (None, 15, 15, 256)
output: (None, 57600)
Dense
input: (None, 57600)
output: (None, 1024)
Dropout
input: (None, 1024)
output: (None, 1024)
Dense
input: (None, 1024)
output: (None, 1024)
Dropout
input: (None, 1024)
output: (None, 1024)
Dense
input: (None, 1024)
output: (None, 4)

Figure A.2: Convolutional Neural Network architecture for the colorectal dataset




Conv2D
input: (None, 65, 81, 64)
output: (None, 63, 79, 64)
MaxPooling2D
input: (None, 63, 79, 64)
output: (None, 31, 39, 64)
Dropout
input: (None, 31, 39, 64)
output: (None, 31, 39, 64)
ZeroPadding2D
input: (None, 31, 39, 64)
output: (None, 33, 41, 64)
Conv2D
input: (None, 33, 41, 64)
output: (None, 31, 39, 128)
ZeroPadding2D
input: (None, 31, 39, 128)
output: (None, 33, 41, 128)
Conv2D
input: (None, 33, 41, 128)
output: (None, 31, 39, 128)
ZeroPadding2D
input: (None, 31, 39, 128)
output: (None, 33, 41, 128)
Conv2D
input: (None, 33, 41, 128)
output: (None, 31, 39, 128)


MaxPooling2D
input: (None, 31, 39, 128)
output: (None, 15, 19, 128)
Dropout
input: (None, 15, 19, 128)
output: (None, 15, 19, 128)
ZeroPadding2D
input: (None, 15, 19, 128)
output: (None, 17, 21, 128)
Conv2D
input: (None, 17, 21, 128)
output: (None, 15, 19, 256)
ZeroPadding2D
input: (None, 15, 19, 256)
output: (None, 17, 21, 256)
Conv2D
input: (None, 17, 21, 256)
output: (None, 15, 19, 256)
ZeroPadding2D
input: (None, 15, 19, 256)
output: (None, 17, 21, 256)
Conv2D
input: (None, 17, 21, 256)
output: (None, 15, 19, 256)
MaxPooling2D
input: (None, 15, 19, 256)
output: (None, 7, 9, 256)


Dropout
input: (None, 7, 9, 256)
output: (None, 7, 9, 256)
Flatten
input: (None, 7, 9, 256)
output: (None, 16128)
Dense
input: (None, 16128)
output: (None, 1024)
Dropout
input: (None, 1024)
output: (None, 1024)
Dense
input: (None, 1024)
output: (None, 1024)
Dropout
input: (None, 1024)
output: (None, 1024)
Dense
input: (None, 1024)
output: (None, 4)

A.2 Networks Training

Figure A.3: Evolution of the loss during training of VGG16 on the prostate dataset using a Xavier weights initialisation

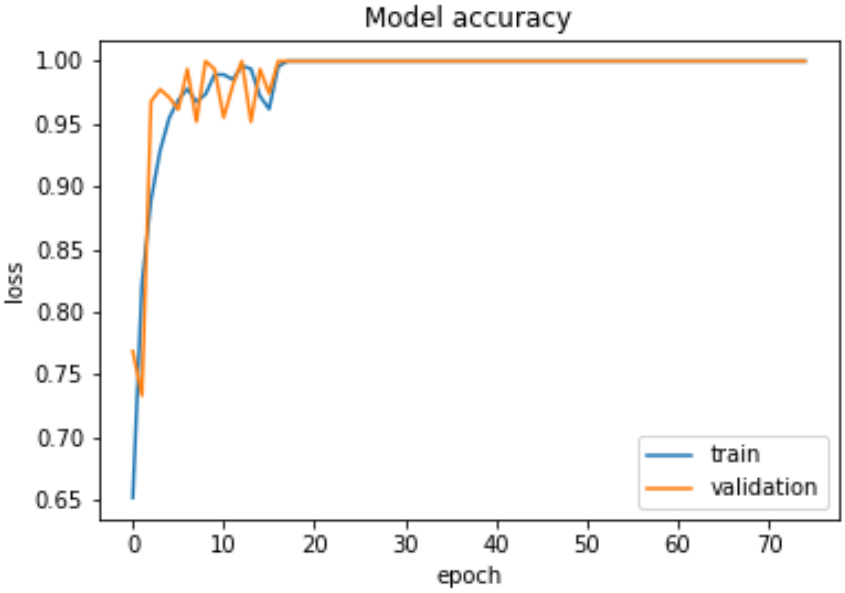


Figure A.4: Evolution of the accuracy during training of VGG16 on the prostate dataset using a Xavier weights initialisation

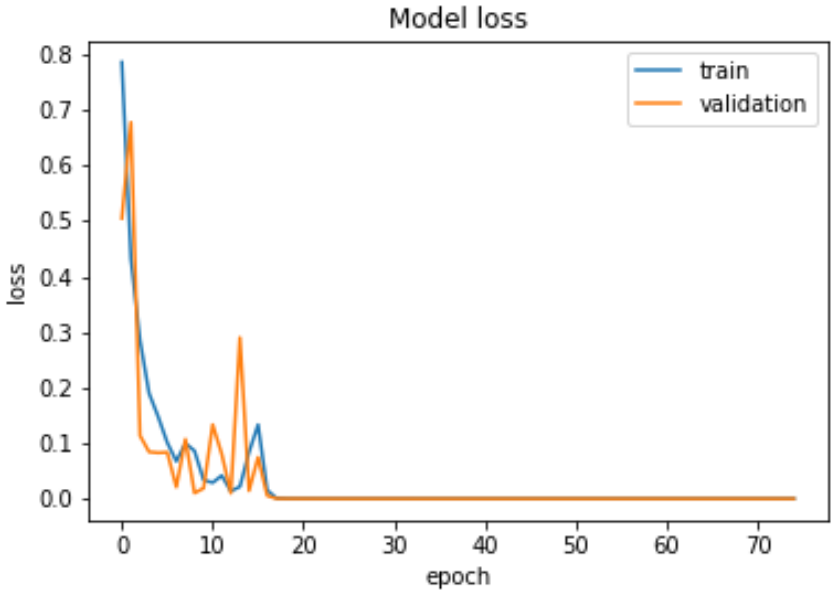


Figure A.5: Evolution of the loss during training of VGG16 on the colorectal dataset using a Xavier weights initialisation

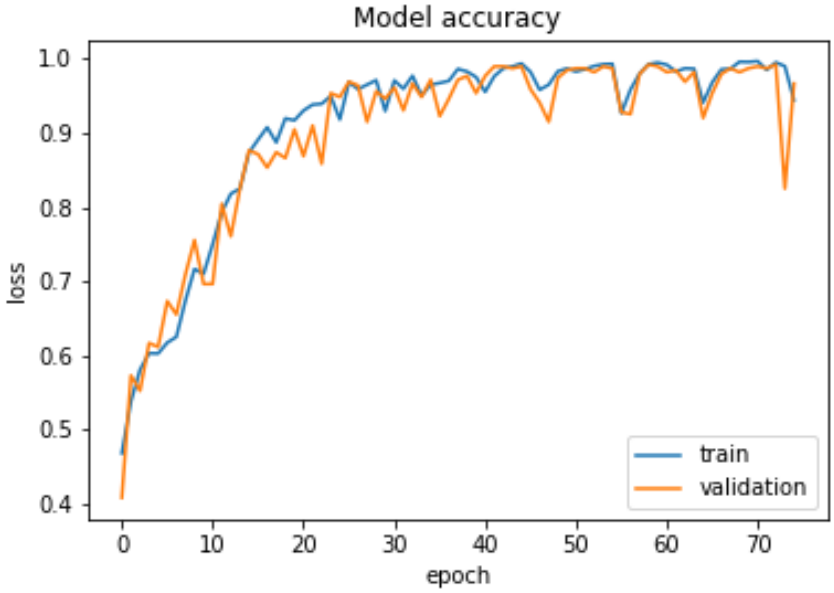


Figure A.6: Evolution of the accuracy during training of VGG16 on the colorectal dataset using a Xavier weights initialisation

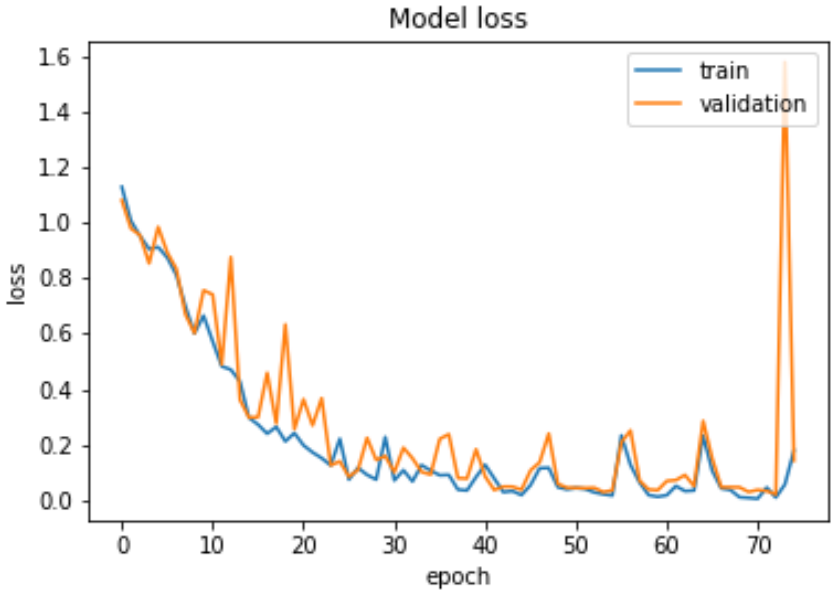


Figure A.7: Evolution of the loss during training of pretrained VGG16 on the prostate dataset

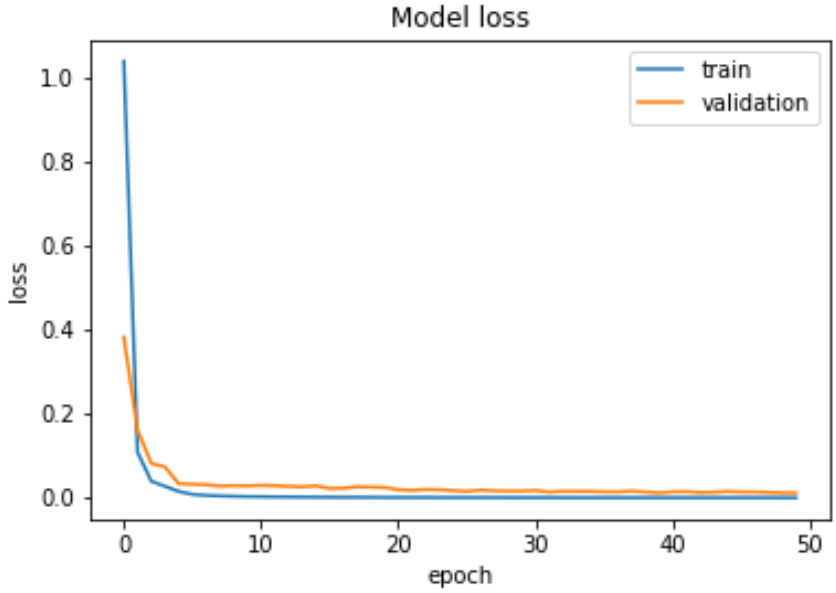


Figure A.8: Evolution of the accuracy during training of pretrained VGG16 on the prostate dataset

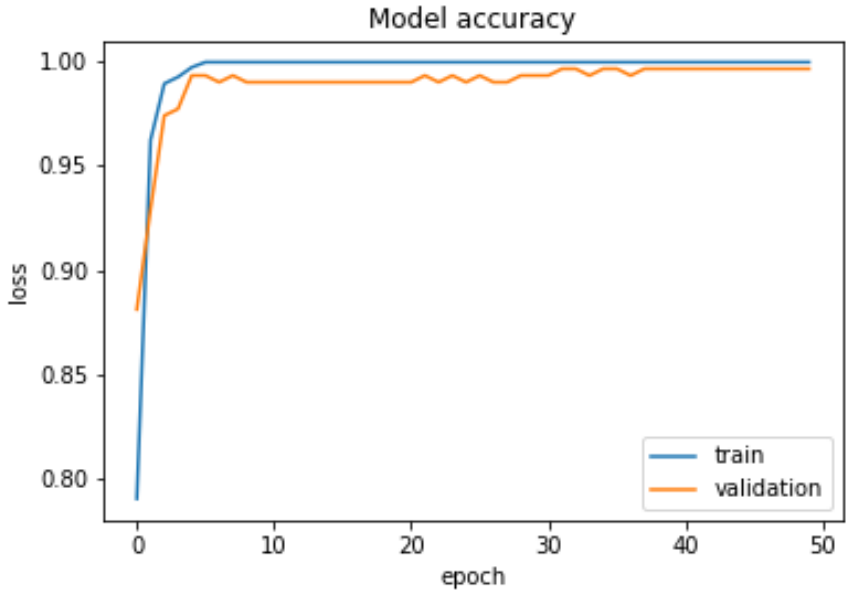


Figure A.9: Evolution of the loss during training of pretrained VGG16 on the colorectal dataset

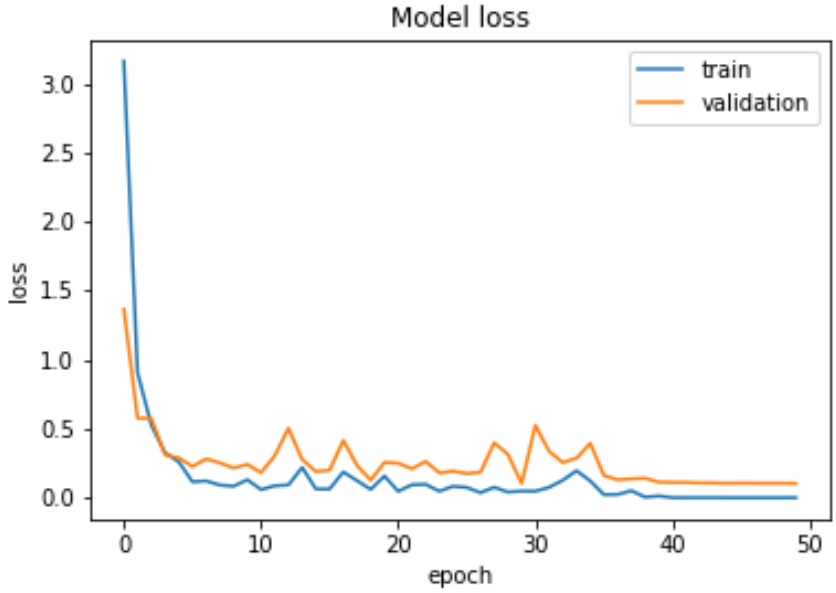


Figure A.10: Evolution of the accuracy during training of pretrained VGG16 on the colorecat dataset

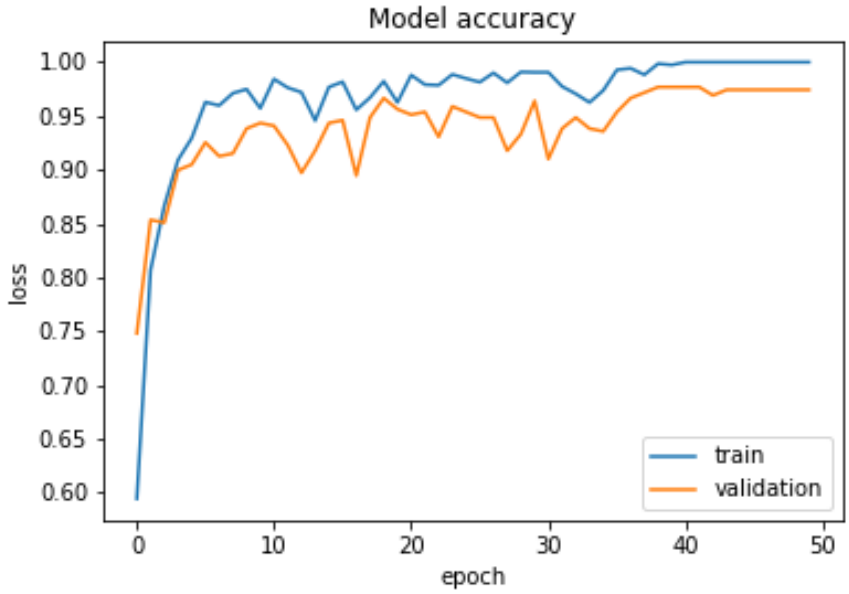


Figure A.11: Evolution of the loss during training of pretrained InceptionV3 on the prostate dataset

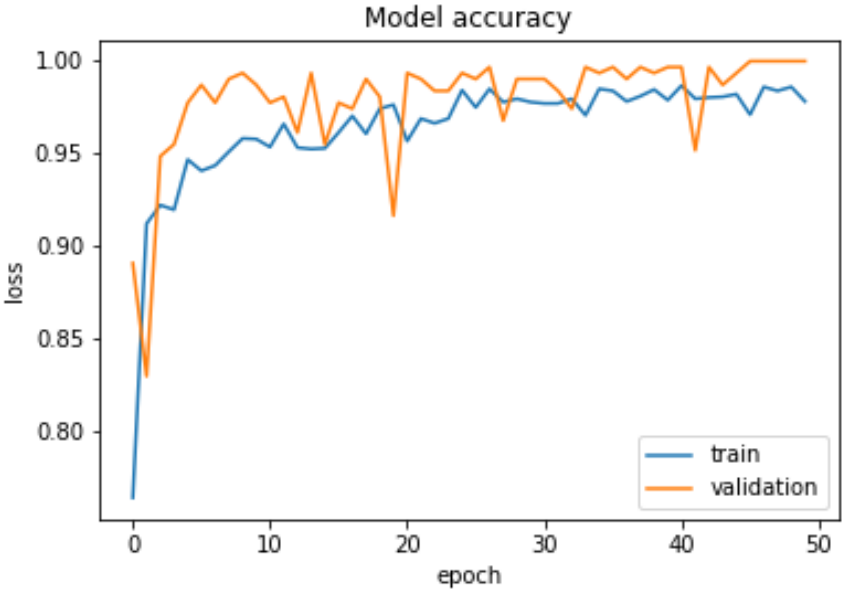


Figure A.12: Evolution of the accuracy during training of pretrained InceptionV3 on the prostate dataset

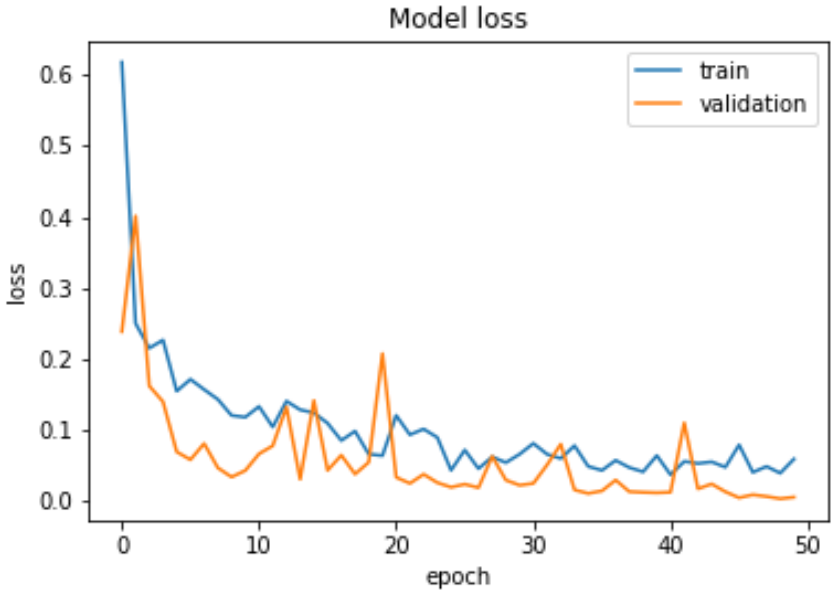


Figure A.13: Evolution of the loss during training of pretrained InceptionV3 on the colorectal dataset

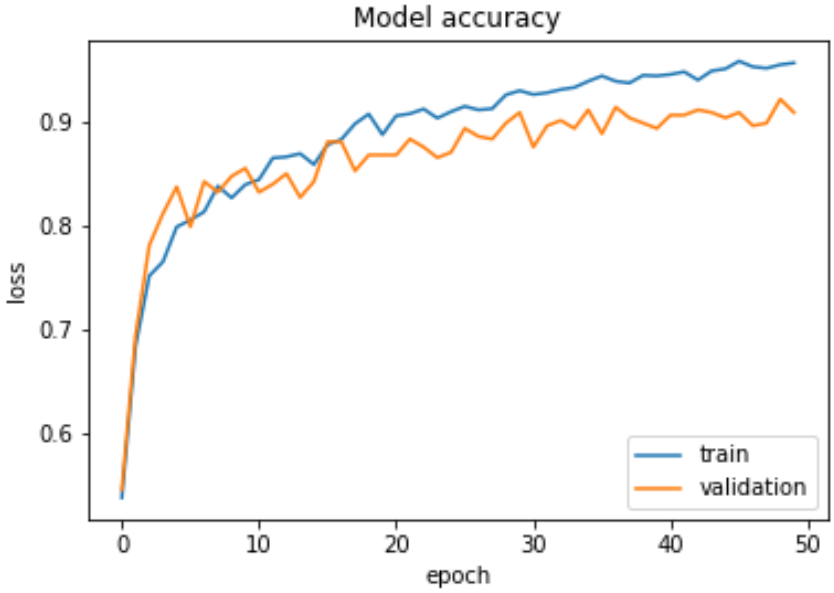


Figure A.14: Evolution of the accuracy during training of pretrained InceptionV3 on the colorectal dataset

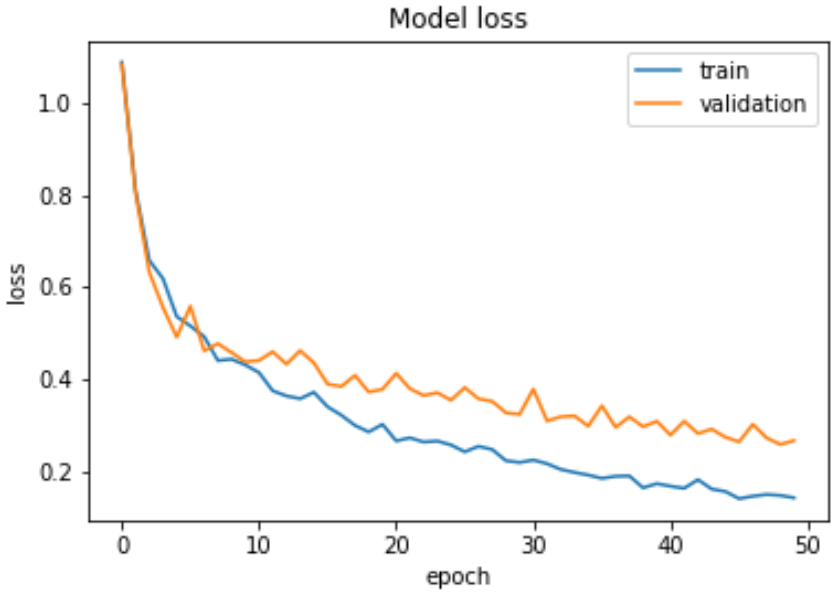


Figure A.15: Evolution of the loss during training of pretrained ResNet50 on the prostate dataset

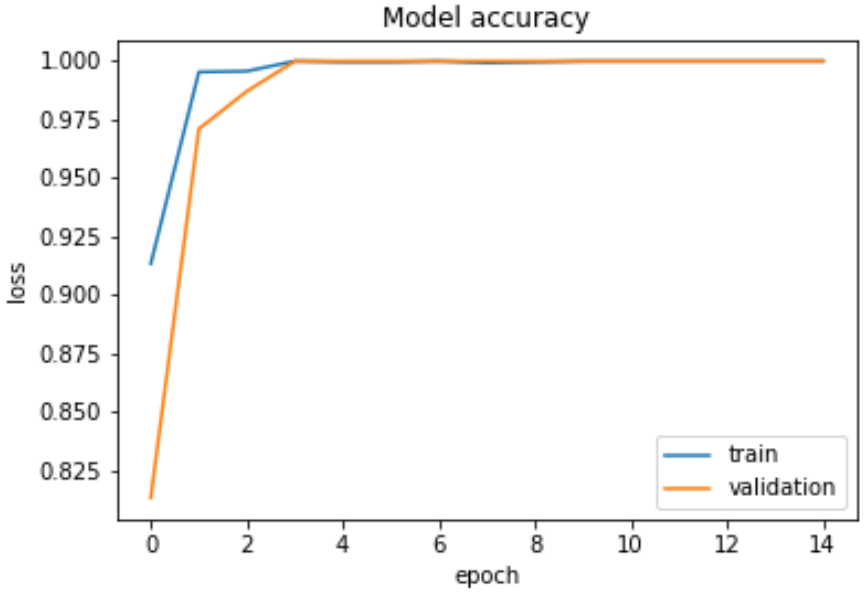


Figure A.16: Evolution of the accuracy during training of pretrained ResNet50 on the prostate dataset

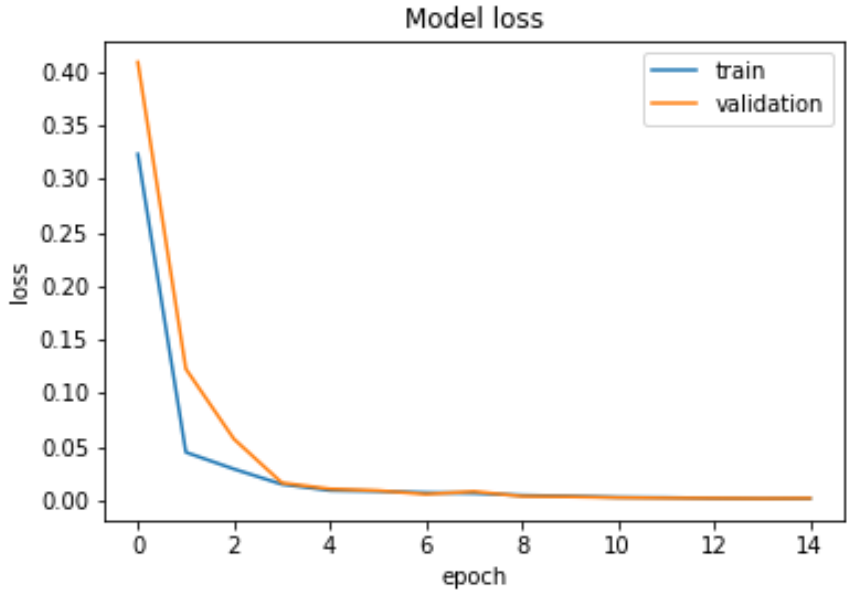


Figure A.17: Evolution of the loss during training of pretrained ResNet50 on the colorectal dataset

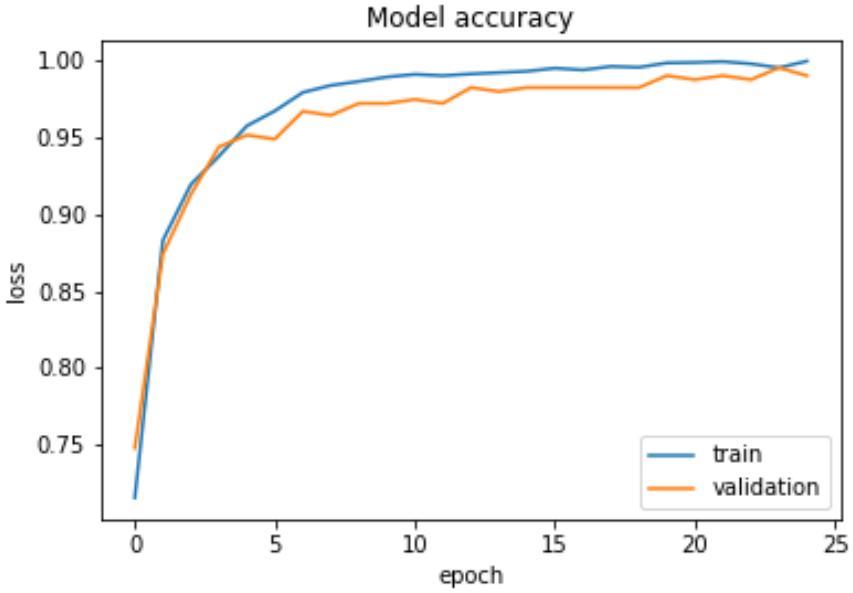
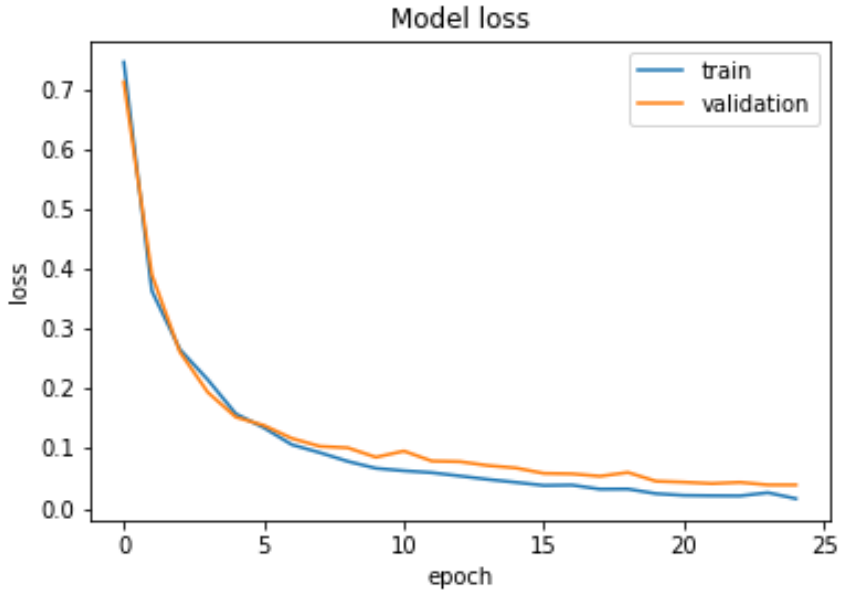


Figure A.18: Evolution of the accuracy during training of pretrained ResNet50 on the colorectal dataset



Bibliography

[1] R. Peyret, A. Bouridane, F. Khelifi, M. A. Tahir, and S. Al-Maadeed, "Automatic classification of colorectal and prostatic histologic tumor images using multiscale multispectral local binary pattern texture features and stacked generalization," *Neurocomputing*, 2017.

[2] R. Peyret, A. Bouridane, S. A. Al-Maadeed, S. Kunhoth, and F. Khelifi, "Texture analysis for colorectal tumour biopsies using multispectral imagery," in *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE*, pp. 7218–7221.

[3] R. Peyret, F. Khelifi, A. Bouridane, and S. Al-Maadeed, "Automatic Diagnosis of Prostate Cancer using Multispectral based Linear Binary Pattern Bagged Codebooks," in *2017 International Conference on Bioengineering for Smart Technologies (BioSMART 2017)*, Aug. 2017.

[4] S. Al Maadeed, S. Kunhoth, A. Bouridane, and R. Peyret, "Multispectral imaging and machine learning for automated cancer diagnosis," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1740–1744, IEEE, June 2017.

[5] J. Ferlay, I. Soerjomataram, M. Ervik, R. Dikshit, S. Eser, C. Mathers, M. Rebelo, D. M. Parkin, D. Forman, and F. Bray, "GLOBOCAN 2012

v.1.1, Cancer Incidence and Mortality Worldwide: IARC CancerBase No. 11 [Internet]," 2014.

[6] A. N. Esgiar, *Texture based computer algorithms for image analysis of colon cancer.* PhD thesis, University of Newcastle upon Tyne, 2000.

[7] A. Heidenreich, J. Bellmunt, M. Bolla, S. Joniau, M. Mason, V. Matveev, N. Mottet, H.-P. Schmid, T. van der Kwast, T. Wiegel, and F. Zattoni, "EAU Guidelines on Prostate Cancer. Part 1: Screening, Diagnosis, and Treatment of Clinically Localised Disease," *European Urology*, vol. 59, no. 1, pp. 61–71, 2011.

[8] M. Tahir, A. Bouridane, and M. Roula, "Prostate cancer classification using multispectral imagery and metaheuristics," *Computational intelligence in medical imaging: techniques and applications, G. Schäfer, A. Hassanien, and J. Jiang, Eds. Taylor & Francis*, pp. 139–166, 2009.

[9] S. Kunhoth, S. Al Maadeed, A. Bouridane, and R. Al Saady, "Medical and Computing Insights into Colorectal Tumors," *International Journal of Life Sciences Biotechnology and Pharma Research*, vol. 4, no. 2, p. 122, 2015.

[10] P. A. Humphrey and Others, *Prostate pathology.* American Society for Clinical Pathology Chicago, 2003.

[11] G. D. Thomas, M. F. Dixon, N. C. Smeeton, and N. S. Williams, "Observer variation in the histological grading of rectal carcinoma.," *Journal of Clinical pathology*, vol. 36, no. 4, pp. 385–391, 1983.

[12] J. D. Kronz, W. H. Westra, and J. I. Epstein, "Mandatory second opinion surgical pathology at a large referral hospital," *Cancer*, vol. 86, no. 11, pp. 2426–2435, 1999.

[13] C. A. Glasbey and G. W. Horgan, *Image analysis for the biological sciences*, vol. 1. Wiley Chichester, 1995.

[14] C. MacAulay and B. Palcic, "Fractal texture features based on optical density surface area. Use in image analysis of cervical cells," *Analytical and quantitative cytology and histology*, vol. 12, Dec. 1990.

[15] C. Mosquera-Lopez, S. Agaian, A. Velez-Hoyos, and I. Thompson, "Computer-Aided Prostate Cancer Diagnosis From Digitized Histopathology: A Review on Texture-Based Systems," *Biomedical Engineering, IEEE Reviews in*, vol. 8, pp. 98–113, 2015.

[16] S. Kunhoth and S. Al Maadeed, "Multispectral Biopsy Image Based Colorectal Tumor Grader," in *Medical Image Understanding and Analysis* (M. Valdés Hernández and V. González-Castro, eds.), vol. 723 of *Communications in Computer and Information Science*, pp. 330–341, Springer International Publishing, 2017.

[17] M. Roula, J. Diamond, A. Bouridane, P. Miller, and A. Amira, "A multispectral computer vision system for automatic grading of prostatic neoplasia," in *Proceedings IEEE International Symposium on Biomedical Imaging*, pp. 193–196, IEEE, 2002.

[18] M. A. Roula, *Machine vision and texture analysis for the automated identification of tissue pattern in prostatic neoplasia*. PhD thesis, 2004.

[19] M. A. Tahir and A. Bouridane, "Novel Round-Robin Tabu Search Algorithm for Prostate Cancer Classification and Diagnosis Using Multispectral Imagery," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 10, no. 4, pp. 782–793, 2006.

[20] M. A. Tahir, A. Bouridane, and F. Kurugollu, "Simultaneous feature selection and feature weighting using Hybrid Tabu Search/K-nearest neighbor classifier," *Pattern Recognition Letters*, vol. 28, no. 4, pp. 438–446, 2007.

[21] S. Bouatmane, M. A. Roula, A. Bouridane, and S. Al-Maadeed, "Round-Robin sequential forward selection algorithm for prostate cancer classification and diagnosis using multispectral imagery," *Machine Vision and Applications*, vol. 22, no. 5, pp. 865–878, 2010.

[22] P. Lasch, L. Chiriboga, H. Yee, and M. Diem, "Infrared Spectroscopy of Human Cells and Tissue: Detection of Disease," *Technology in Cancer Research & Treatment*, vol. 1, no. 1, pp. 1–7, 2002. PMID: 12614171.

[23] S. M. Cohn, E. H. Birnbaum, and C. M. Friel, "Colon: Anatomy and Structural Anomalies," *Textbook of Gastroenterology*, pp. 1369–1385, 2009.

[24] W. Sircus and H. J. Dworken, "Human digestive system," Nov. 2016. accessed on: October 29, 2017.

[25] M. H. Ross, G. I. Kaye, and W. Pawlina, *Histology : a text and atlas : with cell and molecular biology*. Lippincott Williams Wilkins, 2003.

[26] M. J. Miller and R. D. Newberry, "Microanatomy of the intestinal lymphatic system," *Annals of the New York Academy of Sciences*, vol. 1207, pp. E21–E28, Oct. 2010.

[27] J. Lackie, *A Dictionary of Biomedicine*. Oxford University Press, 2010.

[28] J. R. Jass and L. H. Sobin, "Histological Classification of Intestinal Tumours," in *Histological Typing of Intestinal Tumours* (J. R. Jass and L. H. Sobin, eds.), pp. 5–11, Springer Berlin Heidelberg, 1989.

[29] A. of Directors of Anatomic and S. Pathology, "Understanding Your Pathology Report: Colon Polyps (Sessile or Traditional Serrated Adenomas)," 2017.

[30] J. I. Epstein, L. Egevad, M. B. Amin, B. Delahunt, J. R. Srigley, P. A. Humphrey, and Grading Committee, "The 2014 International Society of Urological Pathology (ISUP) Consensus Conference on Gleason Grading of Prostatic Carcinoma: Definition of Grading Patterns and Proposal for a New Grading System.," *The American journal of surgical pathology*, vol. 40, pp. 244–252, Feb. 2016.

[31] D. F. Gleason, "The veteran's administration cooperative urologic research group: Histologic grading and clinical staging of prostatic carcinoma," *Urologic Pathology: The Prostate*, pp. 171–198, 1977.

[32] P. M. Pierorazio, P. C. Walsh, A. W. Partin, and J. I. Epstein, "Prognostic Gleason grade grouping: data based on the modified Gleason scoring system.," *BJU international*, vol. 111, pp. 753–760, May 2013.

[33] D. Kankaya, "Current Status of Histologic Grading in Prostate Carcinoma and Renal Cell Carcinoma," *Journal of Urological Surgery*, vol. 4, pp. 102–105, June 2017.

[34] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
urlhttp://www.deeplearningbook.org.

[35] T. M. Mitchell, "Does machine learning really work?," *AI magazine*, vol. 18, no. 3, p. 11, 1997.

[36] P. Harrington, *Machine learning in action*, vol. 5. Manning Greenwich, CT, 2012.

[37] M. Tuceryan and A. K. Jain, *Texture analysis*, pp. 207–248. World Scientific Publishing, second ed., 1998.

[38] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural Features for Image Classification," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. SMC-3, pp. 610–621, Nov. 1973.

[39] R. Khelifi, M. Adel, and S. Bourennane, "Multispectral texture characterization: application to computer aided diagnosis on prostatic tissue images," *EURASIP Journal on Advances in Signal Processing*, vol. 2012, no. 1, pp. 1–13, 2012.

[40] K. Jafari-Khouzani and H. Soltanian-Zadeh, "Multiwavelet grading of pathological images of prostate," *IEEE Transactions on Biomedical Engineering*, vol. 50, pp. 697–704, June 2003.

[41] J. Diamond, N. H. Anderson, P. H. Bartels, R. Montironi, and P. W. Hamilton, "The use of morphological characteristics and texture analysis in the identification of tissue composition in prostatic neoplasia," *Human Pathology*, vol. 35, pp. 1121–1131, Sept. 2004.

[42] E. Alexandratou, D. Yova, D. Gorpas, P. Maragos, G. Agrogiannis, and N. Kavantzas, "Texture analysis of tissues in Gleason grading of prostate cancer," vol. 6859, pp. 685904–685904–8, 2008.

[43] S. Doyle, M. Feldman, J. Tomaszewski, and A. Madabhushi, "A Boosted Bayesian Multiresolution Classifier for Prostate Cancer Detection From Digitized Needle Biopsies," *IEEE Transactions on Biomedical Engineering*, vol. 59, pp. 1205–1218, May 2012.

[44] K. Nguyen, A. Jain, and B. Sabata, "Prostate cancer detection: Fusion of cytological and textural features," *J Pathol Inform*, vol. 2, p. 3, Dec. 2011.

[45] K. Masood, N. M. Rajpoot, H. A. Qureshi, and K. Rajpoot, "Co-occurrence and morphological analysis for colon tissue biopsy classification," in *4th International Workshop on Frontiers of Information Technology (FIT 2006)*, 2006.

[46] A. Chaddad, C. Desrosiers, A. Bouridane, M. Toews, L. Hassan, and C. Tanougast, "Multi Texture Analysis of Colorectal Cancer Continuum Using Multispectral Imagery," *PLOS ONE*, vol. 11, pp. e0149893+, Feb. 2016.

[47] H. Kalkan, M. Nap, R. P. W. Duin, and M. Loog, "Automated classification of local patches in colon histopathology," in *Pattern Recognition (ICPR), 2012 21st International Conference on*, pp. 61–64.

[48] X. Sun, S.-H. Chuang, J. Li, and F. McKenzie, "Automatic diagnosis for prostate cancer using run-length matrix method," in *Proc. SPIE*, vol. 7260, pp. 72603H–72603H–8, 2009.

[49] A. Tabesh, M. Teverovskiy, H.-Y. Pang, V. P. Kumar, D. Verbel, A. Kotsianti, and O. Saidi, "Multifeature Prostate Cancer Diagnosis and Gleason Grading of Histological Images," *IEEE Transactions on Medical Imaging*, vol. 26, pp. 1366–1378, Oct. 2007.

[50] C. M. Lopez, S. Agaian, I. Sanchez, A. Almuntashri, O. Zinalabdin, A. A. Rikabi, and I. Thompson, "Exploration of efficacy of gland morphology and architectural features in prostate cancer gleason grading," in *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 2849–2854, IEEE, Oct. 2012.

[51] J. N. Kather, C.-A. Weis, F. Bianconi, S. M. Melchers, L. R. Schad, T. Gaiser, A. Marx, and F. G. Zöllner, "Multi-class texture analysis in colorectal cancer histology," *Scientific Reports*, vol. 6, p. 27988, 2016.

[52] K. Masood and N. Rajpoot, "Texture based classification of hyperspectral colon biopsy samples using CLBP," in *Biomedical Imaging: From Nano to Macro, 2009. ISBI '09. IEEE International Symposium on*, pp. 1011–1014.

[53] A. Greenblatt, C. Mosquera-Lopez, and S. Agaian, "Quaternion Neural Networks Applied to Prostate Cancer Gleason Grading," in *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pp. 1144–1149.

[54] E. Ozdemir and C. Gunduz-Demir, "A Hybrid Classification Model for Digital Pathology Using Structural and Statistical Pattern Recognition," *Medical Imaging, IEEE Transactions on*, vol. 32, no. 2, pp. 474–483, 2013.

[55] S. Naik, S. Doyle, S. Agner, A. Madabhushi, M. Feldman, and J. Tomaszewski, "Automated gland and nuclei segmentation for grading of prostate and breast cancer histopathology," in *2008 5th IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 284–287, IEEE, May 2008.

[56] E. Yu, J. P. Monaco, J. Tomaszewski, N. Shih, M. Feldman, and A. Madabhushi, "Detection of prostate cancer on histopathology using color fractals and Probabilistic Pairwise Markov models," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3427–3430, IEEE, Aug. 2011.

[57] A. N. Esgiar, R. N. G. Naguib, B. S. Sharif, M. K. Bennett, and A. Murray, "Fractal analysis in the detection of colonic cancer images," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 6, no. 1, pp. 54–58, 2002.

[58] P.-W. Huang and C.-H. Lee, "Automatic Classification for Pathological Prostate Images Based on Fractal Analysis," *IEEE Transactions on Medical Imaging*, vol. 28, pp. 1037–1050, July 2009.

[59] K. Masood, "Hyperspectral imaging with wavelet transform for classification of colon tissue biopsy samples," vol. 7073, pp. 707319–707319–8, 2008.

[60] A. N. Esgiar, R. N. G. Naguib, B. S. Sharif, M. K. Bennett, and A. Murray, "Microscopic image analysis for quantitative measurement and feature identification of normal and cancerous colonic mucosa," *Information Technology in Biomedicine, IEEE Transactions on*, vol. 2, no. 3, pp. 197–203, 1998.

[61] F. Bianconi, A. A. Larrán, and A. Fernández, "Discrimination Between Tumour Epithelium and Stroma via Perception-based Features," *Neurocomput.*, vol. 154, pp. 119–126, Apr. 2015.

[62] S.-K. Tai, C.-Y. Li, Y.-C. Wu, Y.-J. Jan, and S.-C. Lin, "Classification of prostatic biopsy," in *6th International Conference on Digital Content, Multimedia Technology and its Applications*, pp. 354–358, Aug. 2010.

[63] A. Tabesh, V. P. Kumar, H.-y. Pang, D. Verbel, A. Kotsianti, M. Teverovskiy, and O. Saidi, "58 Automated Prostate Cancer Diagnosis and Gleason Grading of Tissue Microarrays,"

[64] J. P. Monaco, J. E. Tomaszewski, M. D. Feldman, I. Hagemann, M. Moradi, P. Mousavi, A. Boag, C. Davidson, P. Abolmaesumi, and A. Madabhushi, "High-throughput detection of prostate cancer in histological sections using probabilistic pairwise Markov models.," *Medical image analysis*, vol. 14, pp. 617–629, Aug. 2010.

[65] J. Monaco, J. E. Tomaszewski, M. D. Feldman, M. Moradi, P. Mousavi, A. Boag, C. Davidson, P. Abolmaesumi, and A. Madabhushi, "Probabilistic pairwise Markov models: application to prostate cancer detection," vol. 7259, pp. 725903–725903–12, 2009.

[66] J. P. Monaco, J. E. Tomaszewski, M. D. Feldman, M. Moradi, P. Mousavi, A. Boag, C. Davidson, P. Abolmaesumi, and A. Madabhushi, "Detection of Prostate Cancer from Whole-Mount Histology Images Using Markov Random Fields," in *Workshop on Microscopic Image Analysis with Applications in Biology (in conjunction with MICCAI)*, (New York, NY), 2008.

[67] A. Almuntashri, S. Agaian, I. Thompson, D. Rabah, O. Zin Al-Abdin, and M. Nicolas, *Gleason grade-based automatic classification of prostate cancer pathological images*, pp. 2696–2701. 2011.

[68] N. Sengar, N. Mishra, M. K. Dutta, J. Prinosil, and R. Burget, "Grading of colorectal cancer using histology images," in *2016 39th International Conference on Telecommunications and Signal Processing (TSP)*, pp. 529–532, June 2016.

[69] A. Banwari, N. Sengar, M. K. Dutta, and C. M. Travieso, "Automated segmentation of colon gland using histology images," in *2016 Ninth International Conference on Contemporary Computing (IC3)*, pp. 1–5, Aug. 2016.

[70] S. Rathore, M. A. Iftikhar, and M. Hassan, "Ensemble Sparse Classification of Colon Cancer," in *2016 International Conference on Frontiers of Information Technology (FIT)*, pp. 235–240, Dec. 2016.

[71] K. Nguyen, A. K. Jain, and R. L. Allen, "Automated Gland Segmentation and Classification for Gleason Grading of Prostate Tissue Images," in *2010 20th International Conference on Pattern Recognition*, pp. 1497–1500, IEEE, Aug. 2010.

[72] A. Chaddad, C. Tanougast, A. Dandache, A. Al Houseini, and A. Bouridane, "Improving of colon cancer cells detection based on Haralick's features on segmented histopathological images," in *Computer Applications and Industrial Electronics (ICCAIE), 2011 IEEE International Conference on*, pp. 87–90.

[73] S. D. Hilado, L. A. Lim, R. N. Gorgui-Naguib, E. P. Dadios, and J. M. Avila, "Implementation of Wavelets and Artificial Neural Networks in Colonic Histopathological Classification," *JACIII*, vol. 18, no. 5, pp. 792–797, 2014.

[74] R. Khelifi, M. Adel, and S. Bourennane, "Texture classification for multispectral images using spatial and spectral Gray Level Differences," in *2010 2nd International Conference on Image Processing Theory, Tools and Applications*, pp. 330–333, IEEE, July 2010.

[75] R. Khelifi, M. Adel, and S. Bourennane, "Spatial and spectral dependance co-occurrence method for multi-spectral image texture classification," in *2010 IEEE International Conference on Image Processing*, pp. 4361–4364, IEEE, Sept. 2010.

[76] R. Khelifi, M. Adel, and S. Bourennane, *Generalized gray level dependence method for prostate cancer classification.* Algeria: Tipaza, 2011.

[77] M. Hauta-Kasari, J. Parkkinen, T. Jaaskelainen, and R. Lenz, "Generalized co-occurrence matrix for multispectral texture analysis," in *Proceedings of 13th International Conference on Pattern Recognition*, pp. 785–789 vol.2, IEEE, 1996.

[78] A. Chaddad and C. Tanougast, "Texture Analysis of Abnormal Cell Images for Predicting the Continuum of Colorectal Cancer," *Analytical Cellular Pathology*, vol. 2017, pp. 1–13, 2017.

[79] A. Grote, N. S. Schaadt, and F. Feuerhake, "Image analysis approach to distinguish lobular structures in the mammary gland from well-differentiated breast cancer with tubule formation," *The diagnostic pathology journal*, 2016.

[80] H. Irshad, A. Gouaillard, L. Roux, and D. Racoceanu, "Multispectral band selection and spatial characterization: Application to mitosis detection in breast cancer histopathology," *Computerized Medical Imaging and Graphics*, vol. 38, pp. 390–402, July 2014.

[81] G. Zimmerman-Moreno, I. Marin, M. Lindner, I. Barshack, Y. Garini, E. Konen, and A. Mayer, "Automatic classification of cancer cells in multispectral microscopic images of lymph node samples," in *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3973–3976, IEEE, Aug. 2016.

[82] E. Alexandratou, V. Atlamazoglou, T. Thireou, G. Agrogiannis, D. Togas, N. Kavantzas, E. Patsouris, and D. Yova, "Evaluation of Machine Learning Techniques for Prostate Cancer Diagnosis and Gleason Grading," *Int. J. Comput. Intell. Bioinformatics Syst. Biol.*, vol. 1, pp. 297–315, Feb. 2010.

[83] T. C. Smith and E. Frank, *Statistical Genomics: Methods and Protocols*, ch. Introducing Machine Learning Concepts with WEKA, pp. 353–378. New York, NY: Springer, 2016.

[84] J. Ghosh, *Multiple Classifier Systems: Third International Workshop, MCS 2002 Cagliari, Italy, June 2426, 2002 Proceedings*. Lecture Notes in Computer Science 2364, Springer-Verlag Berlin Heidelberg, 1 ed., 2002.

[85] S. Russell and P. Norvig, *Artificial intelligence : a modern approach*. Prentice Hall, 3 ed., Dec. 2010.

[86] S. Doyle, A. Madabhushi, M. Feldman, and J. Tomaszeweski, "A boosting cascade for automated detection of prostate cancer from digitized histology," *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2006*, pp. 504–511, 2006.

[87] S. Doyle, M. Feldman, J. Tomaszewski, N. Shih, and A. Madabhushi, "Cascaded multi-class pairwise classifier (CascaMPa) for normal, cancerous, and cancer confounder classes in prostate histology," in *2011*

IEEE International Symposium on Biomedical Imaging: From Nano to Macro, pp. 715–718, IEEE, Mar. 2011.

[88] S. Doyle, M. D. Feldman, N. Shih, J. Tomaszewski, and A. Madabhushi, "Cascaded discrimination of normal, abnormal, and confounder classes in histopathology: Gleason grading of prostate cancer," *BMC Bioinformatics*, vol. 13, p. 282, 2012.

[89] F. M. Sanghavi and S. S. Agaian, "Automated classification of histopathology images of prostate cancer using a Bag-of-Words approach," vol. 9869, pp. 98690T–98690T–11, 2016.

[90] M. Mirmehdi and M. Petrou, "Segmentation of color textures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, pp. 142–159, Feb. 2000.

[91] H. El Maia, A. Hammouch, and D. Aboutajdine, "Color-texture analysis by mutual information for multispectral image classification," in *2009 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, pp. 359–364, IEEE, Aug. 2009.

[92] C.-H. Chan, J. Kittler, and K. Messer, "Multispectral Local Binary Pattern Histogram for Component-based Color Face Verification," in *2007 First IEEE International Conference on Biometrics: Theory, Applications, and Systems*, pp. 1–7, IEEE, Sept. 2007.

[93] T. Maenpaa, M. Pietikainen, and J. Viertola, "Separating color and pattern information for color texture discrimination," in *Object recognition supported by user interaction for service robots*, pp. 668–671, IEEE Comput. Soc, 2002.

[94] R.-M. Coliban and M. Ivanovici, "Color and multispectral texture characterization using pseudo-morphological tools," in *2014 IEEE International Conference on Image Processing (ICIP)*, pp. 630–634, IEEE, Oct. 2014.

[95] P. Roy Chowdhury, B. Deshmukh, A. K. Goswami, and S. S. Prasad, "Neural Network Based Dunal Landform Mapping From Multispectral Images Using Texture Features," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 4, pp. 171–184, Mar. 2011.

[96] A. Bendada and M. A. Akhloufi, "Multispectral Face Recognition in Texture Space," in *2010 Canadian Conference on Computer and Robot Vision*, pp. 101–106, IEEE, 2010.

[97] H.-I. Kim, S. H. Lee, and Y. M. Ro, "Multispectral Texture Features from Visible and Near-Infrared Synthetic Face Images for Face Recognition," in *2015 IEEE International Symposium on Multimedia (ISM)*, pp. 593–596, IEEE, Dec. 2015.

[98] Y. Zhang, "Texture-integrated classification of urban treed areas in high-resolution color-infrared imagery," in *the First IEEE International Conference on Biometrics: Theory, Applications, and Systems*, vol. 67, pp. 1359–1365, 2001.

[99] M. Smolina and E. Goormaghtigh, "Infrared imaging of MDA-MB-231 breast cancer cell line phenotypes in 2D and 3D cultures," *Analyst*, vol. 140, no. 7, pp. 2336–2343, 2015.

[100] N. Wald and E. Goormaghtigh, "Infrared imaging of primary melanomas reveals hints of regional and distant metastases," *Analyst*, vol. 140, no. 7, pp. 2144–2155, 2015.

[101] R. Wolthuis, A. Travo, C. Nicolet, A. Neuville, M.-P. Gaub, D. Guenot, E. Ly, M. Manfait, P. Jeannesson, and O. Piot, "IR Spectral Imaging for Histopathological Characterization of Xenografted Human Colon Carcinomas," *Analytical Chemistry*, vol. 80, no. 22, pp. 8461–8469, 2008.

[102] T. Ojala, M. Pietikainen, and T. Maenpaa, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 7, pp. 971–987, 2002.

[103] B. F. Zhenhua Wang and F. Wu, "Local Intensity Order Pattern for Feature Description," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 603–610, Nov. 2011.

[104] Y. Bengio, Y. LeCun, and Others, "Scaling learning algorithms towards AI," *Large-scale kernel machines*, vol. 34, no. 5, pp. 1–41, 2007.

[105] W. Härdle, A. Werwatz, M. Müller, and S. Sperlich, "Nonparametric and semiparametric models," 2004.

[106] S. Wold, K. Esbensen, and P. Geladi, "Principal component analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 2, no. 1, pp. 37–52, 1987. Proceedings of the Multivariate Statistical Workshop for Geologists and Geochemists.

[107] I. T. Jolliffe, "Principal Component Analysis and Factor Analysis," in *Principal Component Analysis* (I. T. Jolliffe, ed.), pp. 115–128, Springer New York, 1986.

[108] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, pp. 660–674, May 1991.

[109] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random Forest: A Classification and Regression Tool for Compound Classification and QSAR Modeling," *J. Chem. Inf. Comput. Sci.*, vol. 43, pp. 1947–1958, Nov. 2003.

[110] N. Cristianini and J. Shawe-Taylor, *An Introduction to Support Vector Machines: And Other Kernel-based Learning Methods*. New York, NY, USA: Cambridge University Press, 2000.

[111] C. Burges, "A Tutorial on Support Vector Machines for Pattern Recognition," vol. 2, no. 2, pp. 121–167, 1998.

[112] R. Fletcher, *Quadratic Programming*, pp. 229–258. John Wiley & Sons, Ltd, 2000.

[113] R. Courant and D. Hilbert, "Methods ofMathematical Physics, Vol. I," *Interscience, New York*, pp. 343–350, 1953.

[114] C. Cortes and V. Vapnik, "Support-Vector Networks," *Machine Learning*, vol. 20, pp. 273–297, Sept. 1995.

[115] K. Crammer and Y. Singer, "On the Algorithmic Implementation of Multiclass Kernel-based Vector Machines," *J. Mach. Learn. Res.*, vol. 2, pp. 265–292, Mar. 2002.

[116] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*,

vol. 2, 2011. Software available at
urlhttp://www.csie.ntu.edu.tw/˜cjlin/libsvm.

[117] M. Aly, "Survey on multiclass classification methods," *Technical Report, Caltech*, 2005.

[118] E. L. Allwein, R. E. Schapire, and Y. Singer, "Reducing multiclass to binary: A unifying approach for margin classifiers," *Journal of machine learning research*, vol. 1, no. Dec, pp. 113–141, 2000.

[119] C.-W. Hsu and C.-J. Lin, "A comparison of methods for multiclass support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, pp. 415–425, Mar. 2002.

[120] R. Rifkin and A. Klautau, "In Defense of One-Vs-All Classification," *J. Mach. Learn. Res.*, vol. 5, pp. 101–141, Dec. 2004.

[121] T. Hastie and R. Tibshirani, "Classification by pairwise coupling," in *Advances in neural information processing systems*, pp. 507–513, 1998.

[122] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification (2Nd Edition)*. Wiley-Interscience, 2000.

[123] G. Zhao and M. Pietikainen, "Dynamic Texture Recognition Using Local Binary Patterns with an Application to Facial Expressions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, pp. 915–928, June 2007.

[124] J. Caicedo, A. Cruz, and F. Gonzalez, "Histopathology Image Classification Using Bag of Features and Kernel Functions," in *Artificial Intelligence in Medicine* (C. Combi, Y. Shahar, and A. Abu-Hanna, eds.),

vol. 5651 of *Lecture Notes in Computer Science*, pp. 126–135, Springer Berlin Heidelberg, 2009.

[125] A. Bosch, X. Muñoz, and R. Martí, "Which is the best way to organize/classify images by content?," *Image and Vision Computing*, vol. 25, no. 6, pp. 778–791, 2007.

[126] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," in *Workshop on statistical learning in computer vision, ECCV*, vol. 1, pp. 1–2, Prague, 2004.

[127] J. Sivic, A. Zisserman, and F. Schaffalitzky, "Video google," in *Proceedings of the International Conference on Computer Vision*, vol. 2, pp. 1470–1477, 2003.

[128] T. Tommasi, F. Orabona, and B. Caputo, "CLEF2007 Image Annotation Task: an SVM-based Cue Integration Approach," in *Proceedings of ImageCLEF 2007 -LNCS*, 2007.

[129] D. K. Iakovidis, N. Pelekis, E. E. Kotsifakos, I. Kopanakis, H. Karanikas, and Y. Theodoridis, "A Pattern Similarity Scheme for Medical Image Retrieval," *IEEE Transactions on Information Technology in Biomedicine*, vol. 13, pp. 442–450, July 2009.

[130] L. Breiman and L. Breiman, "Bagging Predictors," in *Machine Learning*, pp. 123–140, 1996.

[131] F. A. Khan, M. A. Tahir, F. Khelifi, A. Bouridane, and R. Almotaeryi, "Robust off-line text independent writer identification using bagged discrete cosine transform features," *Expert Systems with Applications*, vol. 71, pp. 404–415, 2017.

[132] X. Shi, *Independent Component Analysis*, pp. 60–83. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011.

[133] A. Hyvarinen, "Survey on independent component analysis," *Neural computing surveys*, vol. 2, no. 4, pp. 94–128, 1999.

[134] A. Hyvarinen, "Fast and robust fixed-point algorithms for independent component analysis," *IEEE Transactions on Neural Networks*, vol. 10, pp. 626–634, May 1999.

[135] A. Hyvärinen and E. Oja, "Independent component analysis: algorithms and applications," *Neural Networks*, vol. 13, no. 4, pp. 411–430, 2000.

[136] W. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, 1992.

[137] W. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *Bulletin of Mathematical Biology*, vol. 5, pp. 115–133, Dec. 1943.

[138] A. G. Ivakhnenko and V. G. Lapa, "Cybernetics and forecasting techniques," 1967.

[139] K. Fukushima, "Cybernetics 9 by Springer-Verlag 1980 Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position,"

[140] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[141] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Proceedings of the*

25th International Conference on Neural Information Processing Systems, NIPS'12, (USA), pp. 1097–1105, Curran Associates Inc., 2012.

[142] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[143] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," Dec. 2015.

[144] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, S. Dieleman, D. Grewe, J. Nham, N. Kalchbrenner, I. Sutskever, T. Lillicrap, M. Leach, K. Kavukcuoglu, T. Graepel, and D. Hassabis, "Mastering the game of Go with deep neural networks and tree search," *Nature*, vol. 529, pp. 484–489, Jan. 2016.

[145] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, and D. Hassabis, "Mastering the game of Go without human knowledge," *Nature*, vol. 550, pp. 354–359, Oct. 2017.

[146] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, pp. 2278–2324, Nov. 1998.

[147] D. E. Rumelhart, R. Durbin, R. Golden, and Y. Chauvin, "Backpropagation: The basic theory," *Backpropagation: Theory, architectures and applications*, pp. 1–34, 1995.

[148] D. E. Rumelhart, "David E. Rumelhart, Geoffrey E. Hinton, and Ronald J. Williams," *Nature*, vol. 323, pp. 533–536, 1986.

[149] Y. A. LeCun, L. Bottou, G. B. Orr, and K.-R. Müller, "Efficient backprop," in *Neural networks: Tricks of the trade*, pp. 9–48, Springer, 2012.

[150] D. R. Wilson and T. R. Martinez, "The general inefficiency of batch training for gradient descent learning.," *Neural networks : the official journal of the International Neural Network Society*, vol. 16, pp. 1429–1451, Dec. 2003.

[151] P. Y. Simard, D. Steinkraus, and J. Platt, "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis," Institute of Electrical and Electronics Engineers, Inc., Aug. 2003.

[152] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.

[153] D. C. Cireşan, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "High-Performance Neural Networks for Visual Object Classification," Feb. 2011.

[154] D. Cireşan, U. Meier, and J. Schmidhuber, "Multi-column Deep Neural Networks for Image Classification," Feb. 2012.

[155] C. M. Bishop, *Neural networks for pattern recognition*. Oxford university press, 1995.

[156] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B. Delyon, P.-Y. Glorennec, H. Hjalmarsson, and A. Juditsky, "Nonlinear black-box modeling in system identification: a unified overview," *Automatica*, vol. 31, no. 12, pp. 1691–1724, 1995.

[157] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *Journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[158] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," vol. 521, pp. 436–444, May 2015.

[159] Y. T. Zhou, R. Chellappa, A. Vaid, and B. K. Jenkins, "Image restoration using a neural network," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 36, no. 7, pp. 1141–1151, 1988.

[160] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," Apr. 2015.

[161] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.

[162] L. Torrey and J. Shavlik, "Transfer Learning," *IGI Global*, 2009.

[163] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.